

TencentDB for Redis

Announcements

Product Documentation



Copyright Notice

©2013-2024 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 Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. 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

Announcements

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

Announcements

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
<code>DescribeInstanceMonitorBigKey</code>	Queries the big key of an instance
<code>DescribeInstanceMonitorBigKeySizeDist</code>	Queries the big key size distribution of an instance
<code>DescribeInstanceMonitorBigKeyTypeDist</code>	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) instance
Used Memory	storage_min	Used Memory	mem_used	MB	Actually used memory capacity, including capacity for data cache
Capacity	storage_us_min	Memory	mem_util	%	The ratio of the a

Utilization		Utilization			used memory to requested total m
Max Capacity Utilization	StorageMaxUsMin	Max Memory Utilization of Node	mem_max_util	%	The maximum ar all node (shard o 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 a used private incb traffic to the maxi 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 a used private outtk traffic to the maxi traffic
Connections	connections_min	Connections	connections	-	The number of T connections to a instance
Connection Utilization	connections_us_min	Connection Utilization	connections_util	%	The ratio of the n of TCP connectic the maximum num of connections
Slow Logs	slow_query_min	Slow Logs	cmd_slow	-	The number of command execu with a latency gre than the <code>slowlog-slower-t</code> configuration
Total Keys	keys_min	Total Keys	keys	-	The total number keys (level-1 key

					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 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 server
Average Write Latency	latency_set_min	Average Write Latency	latency_write	ms	The average execution latency of write command between 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 execution

					per second
Write Requests	stat_set_min	Write Requests	cmd_write	Counts/sec	The number of w command execut per second
Other Requests	stat_other_min	Other Requests	cmd_other	Counts/sec	The number of command (exclu write and read commands) exec per second
Big Value Requests	big_value_min	Big Value Requests	cmd_big_value	Counts/sec	The number of executions of req larger than 32 KE second
Read Request Hits	stat_success_min	Read Request Hits	cmd_hits	-	The number of ke successfully requ by read comman which is equal to value of the <code>keyspace_hi</code> metric output by <code>info</code> comman
Read Request Misses	stat_missed_min	Read Request Misses	cmd_miss	-	The number of ke unsuccessfully requested by rea commands, whic equal to the value <code>keyspace_mi</code> metric output by <code>info</code> comman
Execution Errors	cmd_err_min	Execution Errors	cmd_err	-	The number of command execut errors. For exampl the command do exist, parameters incorrect, etc.
Read Request Hit Rate	cache_hit_ratio_min	Read Request Hit Rate	cmd_hits_ratio	%	Key hits/(key hits misses). This me can reflect the sit of cache miss. W

					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 utilization of a (shard or replica) instance
Used Memory	storage_min	Used Memory	mem_used	MB	Actually used memory capacity, including capacity for data cache
Memory Utilization	storage_us_min	Memory Utilization	mem_util	%	The ratio of the 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 all node (shard or replica) memory utilizations in an instance
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 used private network traffic to the maximum traffic
Outbound	out_flow_min	Outbound	out_flow	MB/s	Private network

Traffic		Traffic			outbound traffic
Outbound Traffic Utilization	out_flow_us_min	Outbound Traffic Utilization	out_bandwidth_util	%	The ratio of the used private outbound traffic to the maximum traffic
Connections	connections_min	Connections	connections	-	The number of connections to instance
Connection Utilization	connections_us_min	Connection Utilization	connections_util	%	The ratio of the used of TCP connections to the maximum number of connections
Slow Logs	slow_query_min	Slow Logs	cmd_slow	-	The number of command execution with a latency greater than the <code>slow-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 expired keys in a time window, which is related 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 evicted keys in a time window, which is related 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 proxy and the server

Average Read Latency	latency_get_min	Average Read Latency	latency_read	ms	The average latency of read commands between the proxy and the server
Average Write Latency	latency_set_min	Average Write Latency	latency_write	ms	The average latency of write command between proxy and the server
Average Latency of Other Commands	latency_other_min	Average Latency of Other Commands	latency_other	ms	The average latency of command (excluding write and read command) 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 executed per second
Write Requests	stat_set_min	Write Requests	cmd_write	Counts/sec	The number of write command executed per second
get Requests	cmdstat_get_min	Other Requests	cmd_other	Counts/sec	The number of other command (excluding write and read commands) executed 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_setnx_min				
Big Value Requests	big_value_min	Big Value Requests	cmd_big_value	Counts/sec	The number of executions of r larger than 32 second
Read Request Hits	stat_success_min	Read Request Hits	cmd_hits	-	The number of successfully re by read comm which is equal value of the <code>keyspace_1</code>

					metric output k info comn
Read Request Misses	stat_missed_min	Read Request Misses	cmd_miss	-	The number of unsuccessfully requested by r commands, wl equal to the va keyspace_ metric output k info comn
Execution Errors	cmd_err_min	Execution Errors	cmd_err	-	The number of command exe errors. For exa the command exist, paramet incorrect, etc.
Read Request Hit Rate	cache_hit_ratio_min	Read Request Hit Rate	cmd_hits_ratio	%	Key hits/(key h misses). This i can reflect the of cache miss. the access rec quantity is 0, tl of this metric v 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	StorageMin	Used	MemUsed	MB	Actually used memc

Memory		Memory			capacity, including the capacity for data and cache
Memory Utilization	StorageUsMin	Memory Utilization	MemUtil	%	The ratio of the actual 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	InBandwidthUtil	%	The ratio of the actual used private inbound traffic to the maximum traffic
Outbound Traffic	OutFlowMin	Outbound Traffic	OutFlow	MB/s	Private network outbound traffic

Outbound Traffic Utilization	OutFlowUsMin	Outbound Traffic Utilization	OutBandwidthUtil	%	The ratio of the actual 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 proxy and the Redis server
Average Latency of Other Commands	LatencyOtherMin	Average Latency of Other Commands	LatencyOther	ms	The average execution latency of command (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 execution per second
Write Requests	StatSetMin	Write Requests	CmdWrite	Counts/sec	The number of write command execution per second
Other Requests	StatOtherMin	Other Requests	CmdOther	Counts/sec	The number of command (excluding write and read commands) execution per second
Big Value	BigValueMin	Big Value	CmdBigValue	Counts/sec	The number of

Requests		Requests			executions of requests larger than 32 KB per second
Slow Logs	SlowQueryMin	Slow Logs	CmdSlow	-	The number of command execution 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 + 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 a shards in a cluster
Used Memory	StorageMin	Used Memory	MemUsed	MB	Actually used memc capacity, including t capacity for data an cache
Memory Utilization	StorageUsMin	Memory Utilization	MemUtil	%	The ratio of the actu used memory to the requested total men
Max Memory Utilization of Shard	StorageMaxUsMin	Max Memory Utilization of Node	MemMaxUtil	%	The highest memory utilization value of a shards in a cluster
Total Keys	KeysMin	Total Keys	Keys	-	The total number of keys (level-1 keys) i instance storage
Expired Keys	ExpiredKeysMin	Expired Keys	Expired	-	The number of keys expired in a time window, which is eq 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 eq 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 num of TCP connections the maximum numb of connections
Inbound Traffic	InFlowMin	Inbound Traffic	InFlow	MB/s	Private network inbound traffic
Inbound Traffic Utilization	InFlowUsMin	Inbound Traffic Utilization	InBandwidthUtil	%	The ratio of the actu used private inbound traffic to the maximu traffic
Outbound Traffic	OutFlowMin	Outbound Traffic	OutFlow	MB/s	Private network outbound traffic
Outbound Traffic Utilization	OutFlowUsMin	Outbound Traffic Utilization	OutBandwidthUtil	%	The ratio of the actu used private outboud traffic to the maximu traffic
Average Execution Latency	LatencyMin	Average Execution Latency	LatencyAvg	ms	The average execut latency between the proxy and the Redis server
Average Read Latency	LatencyGetMin	Average Read Latency	LatencyRead	ms	The average execut latency of read commands between the proxy and the Redis server
Average Write Latency	LatencySetMin	Average Write Latency	LatencyWrite	ms	The average execut latency of write command between proxy and the Redis server
Average Latency of Other Commands	LatencyOtherMin	Average Latency of Other Commands	LatencyOther	ms	The average execut latency of commanc (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 execution per second
Write Requests	StatSetMin	Write Requests	CmdWrite	Counts/sec	The number of write command execution per second
Other Requests	StatOtherMin	Other Requests	CmdOther	Counts/sec	The number of command (excluding write and read commands) execution 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 execution 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 <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 + 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 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 key 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 execution per second
Write Requests	StatSetNodeMin	Write Requests	CmdWriteNode	Counts/sec	The number of write command execution per second
Other Requests	StatOtherNodeMin	Other Requests	CmdOtherNode	Counts/sec	The number of other command (excluding write and read commands) execution per second
Slow Logs	SlowQueryNodeMin	Slow Logs	CmdSlowNode	-	The number of command execution with a latency greater than the <code>slowlog-log-slower-than</code> configuration
Read Request Hits	StatSuccessNodeMin	Read Request Hits	CmdHitsNode	-	The number of key 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 key unsuccessfully requested by read

					commands, which equal to the value of <code>keyspace_mis</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 + 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 Granularity)	Dimension Name	Description	Format	Mapping
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 DescribeInstances 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 DescribeInstanceNodeInfo 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

Configure Alarm

☒ Cloud Database / Redis / Memory Edition / Instance

☐ Cloud Database / Redis / Memory Edition / Proxy

☐ Cloud Database / Redis / Memory Edition / Redis Node

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.