

TencentDB for MySQL

백서

제품 문서



Tencent Cloud

Copyright Notice

©2013-2023 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.

목录:

백서

성능 백서

성능 테스트 보고서

성능 개요

테스트 방법

테스트 환경

테스트 툴

테스트 방법

테스트 지표

테스트 결과

MySQL 8.0 테스트 결과

MySQL 5.7 테스트 결과

MySQL 5.6 테스트 결과

네트워크 아키텍처 성능 비교

보안 백서

개요

데이터 스토리지 보안

액세스 제어

데이터 통신 보안

데이터 재해 복구

백서

성능 백서

성능 테스트 보고서

성능 개요

최종 업데이트 날짜: : 2022-03-31 15:08:06

TencentDB for MySQL은 오픈 소스 데이터베이스 MySQL을 기반으로 Tencent Cloud가 특별히 구축한 고성능 분산형 데이터 스토리지 서비스로, 사용자가 클라우드에서 관계형 데이터베이스를 보다 쉽게 설정, 운영 및 확장할 수 있습니다. TencentDB for MySQL은 자체 개발한 TXSQL 커널을 사용하여 고성능과 높은 안정성을 제공합니다. 자체 개발한 커널은 엔터프라이즈급 TDE, 감사, 스레드 풀 등 많은 핵심 기능을 갖추고 있어 CDB의 성능과 안정성이 크게 향상됩니다.

Tencent Cloud 데이터베이스 MySQL은 전문팀에 의해 지속적으로 테스트 및 최적화되었으며 유연하고 고효율적인 트랜잭션 처리 능력과 고급 및 완벽한 컴플라이언스 보안 및 데이터 암호화 능력을 갖추고 있어 제품 보안성 및 비즈니스 능력을 크게 보장합니다.

TencentDB for MySQL의 성능 테스트 결과는 다음을 참고하십시오.

- [MySQL 8.0테스트 결과](#)
- [MySQL 5.7테스트 결과](#)
- [MySQL 5.6테스트 결과](#)

테스트 방법

테스트 환경

최종 업데이트 날짜: : 2022-03-31 15:08:06

본 문서는 TencentDB for MySQL 성능 테스트에 사용되는 환경을 소개합니다.

- 리전역/가용존: 베이징 - 베이징 7존
- 클라이언트: S5.8XLARGE64(표준형S5, 32코어 64GB)
- 클라이언트 운영 체제: CentOS 8.2 64비트
- 네트워크: 클라우드 서버(CVM) 및 TencentDB for MySQL 인스턴스의 네트워크 유형은 모두 사설 네트워크(VPC)이며 동일한 서브넷 아래에 있습니다.

테스트를 위한 TencentDB for MySQL 인스턴스 정보는 다음과 같습니다.

- 스토리지 유형: 로컬 SSD 디스크
- 인스턴스 사양: 범용형
- 매개변수 템플릿: 고성능 템플릿

테스트 툴

최종 업데이트 날짜: : 2022-03-31 15:08:06

본 문서에서는 TencentDB for MySQL 성능 테스트 툴인 SysBench와 CVM 인스턴스에 SysBench를 설치하는 방법을 소개합니다.

SysBench 툴 소개

SysBench는 고부하 데이터베이스를 실행할 때 시스템의 관련 핵심 매개변수의 성능을 평가하는 데 사용되는 크로스 플랫폼 및 멀티 스레드 모듈식 벤치마크 테스트 툴입니다. 복잡한 데이터베이스 벤치마크 설정을 무시하고 데이터베이스가 설치되지 않은 경우에도 데이터베이스 시스템 성능을 빠르게 알 수 있습니다.

SysBench 테스트 모델

- SysBench 표준 OLTP 읽기/쓰기 혼합 시나리오에서 트랜잭션에는 18개의 읽기/쓰기 SQL이 포함됩니다.
- SysBench 표준 OLTP 읽기 전용 시나리오에서 트랜잭션에는 14개의 읽기 SQL(10개의 기본 키 포인트 쿼리, 4개의 범위 쿼리)이 포함됩니다.
- SysBench 표준 OLTP 쓰기 전용 시나리오에서 트랜잭션에는 4개의 쓰기 SQL(2개의 UPDATE, 1개의 DELETE 및 1개의 INSERT)이 포함됩니다.

SysBench 매개변수 설명

매개변수	설명
db-driver	데이터베이스 엔진
mysql-host	MySQL 인스턴스 연결 주소
mysql-port	MySQL 인스턴스 연결 포트
mysql-user	MySQL 인스턴스 계정
mysql-password	MySQL 인스턴스 계정에 해당하는 비밀번호
mysql-db	MySQL 인스턴스 데이터베이스 이름

매개변수	설명
table_size	테스트 테이블 크기
tables	테스트 테이블 수
events	테스트 요청 수
time	테스트 시간
threads	테스트 스레드 수
percentile	통계할 백분율, 기본값은 95%, 즉 95%의 경우에서 실행 시간 요청
report-interval	테스트 진행 보고서가 N초에 한 번 출력됨을 나타내며, 0은 테스트 진행 보고서 출력이 비활성화 되고 최종 보고서 결과만 출력됨을 의미
skip-trx	트랜잭션 건너뛰기 여부. 1: 건너뛰기 0: 건너뛰지 않음

설치 방법

이 부하 테스트는 SysBench 버전 1.0.20을 사용합니다. 자세한 내용은 [Sysbench 공식 문서](#)를 참고하십시오.

1. CVM 인스턴스에서 다음 명령어를 실행하여 SysBench를 설치합니다.

```

yum install gcc gcc-c++ autoconf automake make libtool bzip2 mysql-devel git mysql
l
git clone https://github.com/akopytov/sysbench.git
##Git에서 SysBench 다운로드
cd sysbench
##SysBench 디렉터리 열기
git checkout 1.0.20
##SysBench 1.0.20 버전 으로 전환
./autogen.sh
##autogen.sh 실행
./configure --prefix=/usr --mandir=/usr/share/man
make
##컴파일
make install

```

2. 다음 명령을 실행하여 커널이 모든 CPU를 사용하여 데이터 패킷을 처리하는 동시에 CPU 간의 컨텍스트 전환을 줄일 수 있도록 클라이언트를 설정합니다.

```
sudo sh -c 'for x in /sys/class/net/eth0/queues/rx-*; do echo ffffffff>$x/rps_cpus; done'
sudo sh -c "echo 32768 > /proc/sys/net/core/rps_sock_flow_entries"
sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-0/rps_flow_cnt"
sudo sh -c "echo 4096 > /sys/class/net/eth0/queues/rx-1/rps_flow_cnt"
```

설명 :

fffffff는 32개의 CPU 사용을 의미합니다(1개 f는 4개의 CPU를 의미).

테스트 방법

최종 업데이트 날짜: : 2022-03-31 15:08:06

본 문서에서는 TencentDB for MySQL 성능 테스트 방법을 소개합니다.

작업 단계

SysBench를 사용하여 TencentDB for MySQL 인스턴스의 읽기/쓰기 하이브리드 성능을 테스트합니다.

1. 데이터를 준비합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX --mysql-user=XXX -  
-mysql-password=XXX --mysql-db=sbtest --table_size=25000 --tables=250 --events=  
0 --time=600 oltp_read_write prepare
```

2. workload를 실행합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX --mysql-user=XXX -  
-mysql-password=XXX --mysql-db=sbtest --table_size=25000 --tables=250 --events=  
0 --time=600 --threads=XXX --percentile=95 --report-interval=1 oltp_read_write  
run
```

3. 데이터를 제거합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX --mysql-user=XXX -  
-mysql-password=XXX --mysql-db=sbtest --table_size=25000 --tables=250 --events=  
0 --time=600 --threads=XXX --percentile=95 oltp_read_write cleanup
```

테스트 지표

최종 업데이트 날짜: : 2022-03-31 15:08:06

본 문에서는 TencentDB for MySQL 성능 테스트의 테스트 지표를 소개합니다.

테스트 지표

- **초당 실행된 트랜잭션 수 TPS(Transactions Per Second)**

데이터베이스의 초당 트랜잭션 수는 COMMIT 성공 횟수를 기준으로 합니다.

- **초당 실행된 요청 수 QPS(Queries Per Second)**

INSERT, SELECT, UPDATE, DELETE, COMMIT 등을 포함하여 데이터베이스에서 초당 실행된 SQL 수입니다.

- **모든 event의 평균 소요 시간 avg_lat(Average Latency)**

테스트 결과

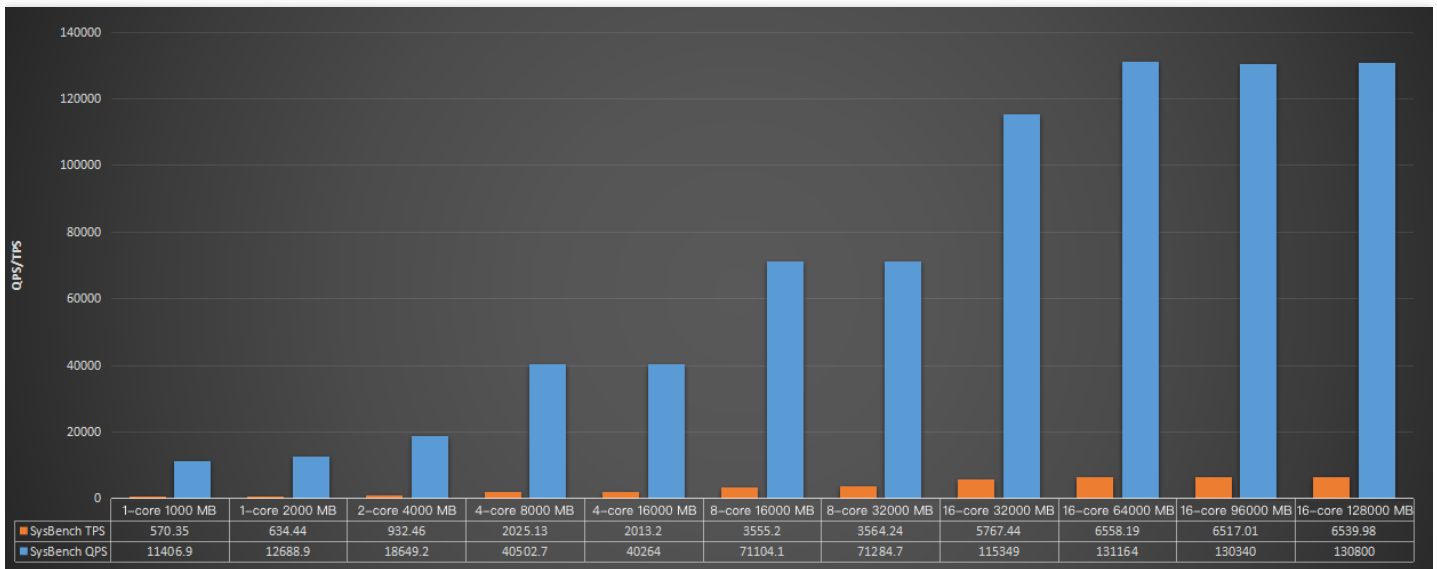
MySQL 8.0 테스트 결과

최종 업데이트 날짜: : 2022-03-31 15:08:06

본 문에서는 TencentDB for MySQL 8.0 범용 인스턴스용 성능 테스트 결과를 소개합니다.

시나리오1: 전체 캐시

전체 캐시는 모든 데이터만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하기 위해 디스크를 읽고 쓸 필요가 없음을 의미합니다.

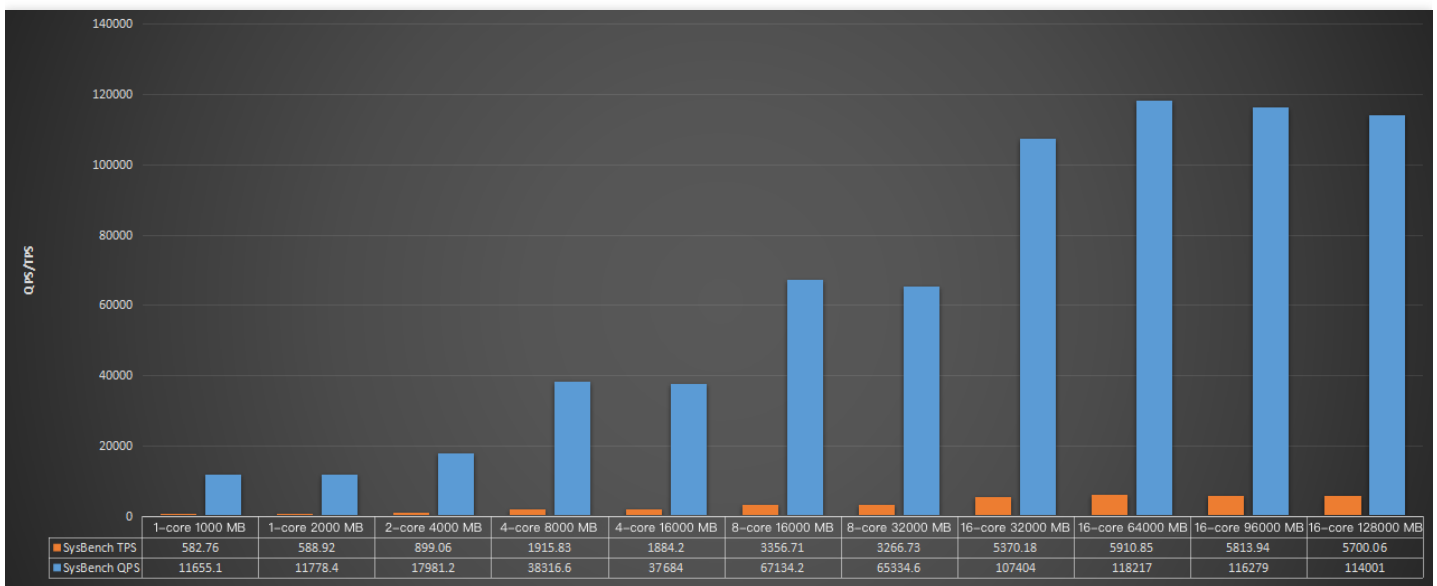


CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	25000	150	570.35	11406.9	14.03
1	2000	8	25000	150	634.44	12688.9	12.61
2	4000	16	25000	150	932.46	18649.2	17.16
4	8000	32	25000	150	2025.13	40502.7	15.8
4	16000	32	25000	150	2013.2	40264	15.89
8	16000	64	25000	150	3555.2	71104.1	18

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
8	32000	64	25000	150	3564.24	71284.7	17.95
16	32000	128	25000	150	5767.44	115349	22.19
16	64000	128	25000	150	6558.19	131164	19.51
16	96000	128	25000	150	6517.01	130340	19.63
16	128000	128	25000	150	6539.98	130800	19.57

시나리오2: 디스크 IO 유형

디스크 IO 유형 시나리오는 데이터의 일부만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하려면 디스크를 읽기/쓰기 해야 합니다.



CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	800000	6	582.76	11655.1	13.73
1	2000	8	800000	12	588.92	11778.4	13.58
2	4000	16	800000	24	899.06	17981.2	17.8
4	8000	32	800000	48	1915.83	38316.6	16.7

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
4	16000	32	6000000	13	1884.2	37684	16.98
8	16000	64	6000000	13	3356.71	67134.2	19.06
8	32000	64	6000000	25	3266.73	65334.6	19.59
16	32000	128	6000000	25	5370.18	107404	23.83
16	64000	128	6000000	49	5910.85	118217	21.65
16	96000	128	6000000	74	5813.94	116279	22.01
16	128000	128	6000000	98	5700.06	114001	22.45

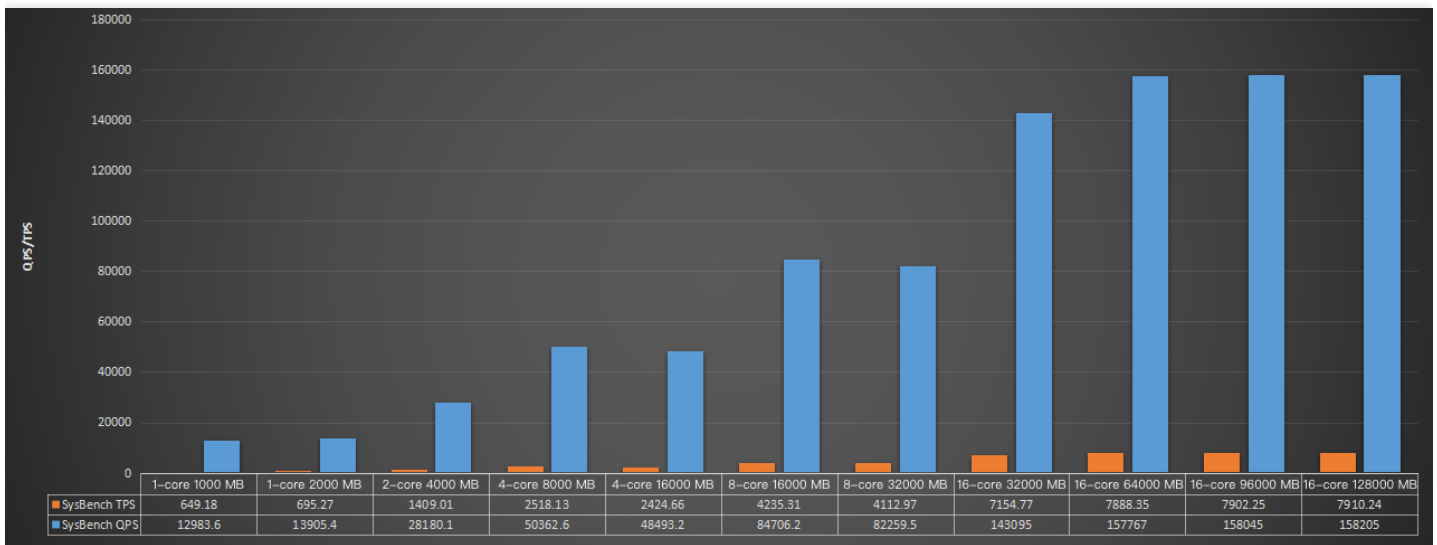
MySQL 5.7 테스트 결과

최종 업데이트 날짜 : 2022-03-31 15:08:06

본 문에서는 TencentDB for MySQL 5.7 범용 인스턴스용 성능 테스트 결과를 소개합니다.

시나리오1: 전체 캐시

전체 캐시는 모든 데이터만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하기 위해 디스크를 읽고 쓸 필요가 없음을 의미합니다.

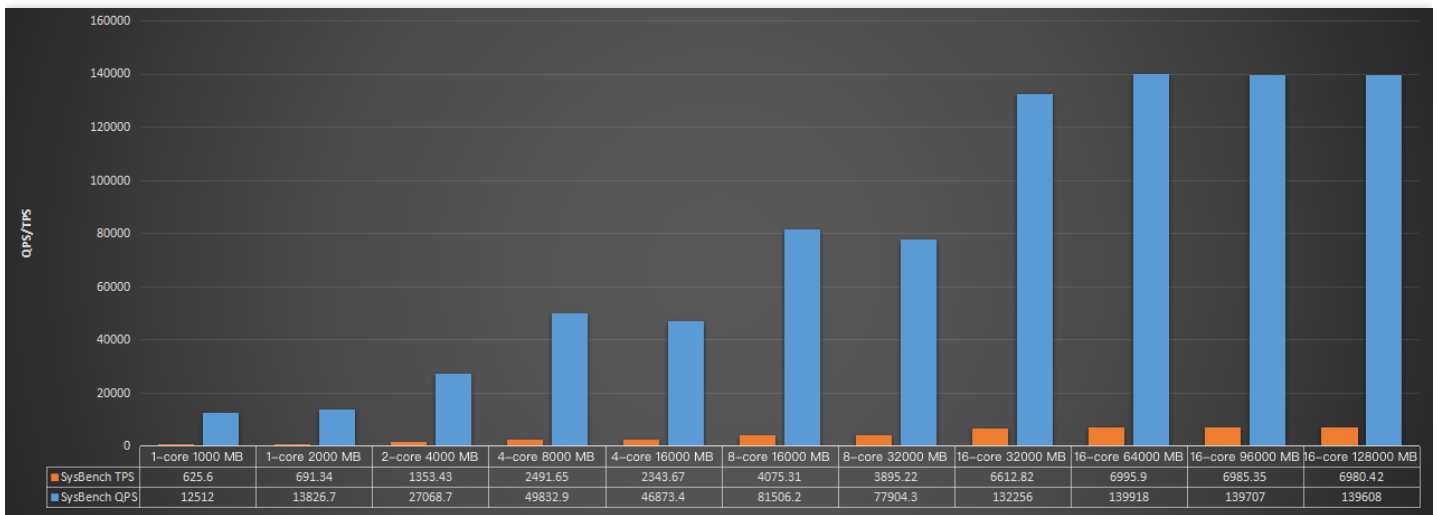


CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	25000	150	649.18	12983.6	12.32
1	2000	8	25000	150	695.27	13905.4	11.51
2	4000	16	25000	150	1409.01	28180.1	11.35
4	8000	32	25000	150	2518.13	50362.6	12.71
4	16000	32	25000	150	2424.66	48493.2	13.2
8	16000	64	25000	150	4235.31	84706.2	15.11
8	32000	64	25000	150	4112.97	82259.5	15.56
16	32000	128	25000	150	7154.77	143095	17.88

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
16	64000	128	25000	150	7888.35	157767	16.22
16	96000	128	25000	150	7902.25	158045	16.19
16	128000	128	25000	150	7910.24	158205	16.17

시나리오2: 디스크 IO 유형

디스크 IO 유형 시나리오는 데이터의 일부만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하려면 디스크를 읽기/쓰기 해야 합니다.



CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	800000	6	625.6	12512	12.79
1	2000	8	800000	12	691.34	13826.7	11.57
2	4000	16	800000	24	1353.43	27068.7	11.82
4	8000	32	800000	48	2491.65	49832.9	12.84
4	16000	32	6000000	13	2343.67	46873.4	13.65
8	16000	64	6000000	13	4075.31	81506.2	15.7
8	32000	64	6000000	25	3895.22	77904.3	16.43

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
16	32000	128	6000000	25	6612.82	132256	19.35
16	64000	128	6000000	49	6995.9	139918	18.29
16	96000	128	6000000	74	6985.35	139707	18.32
16	128000	128	6000000	98	6980.42	139608	18.33

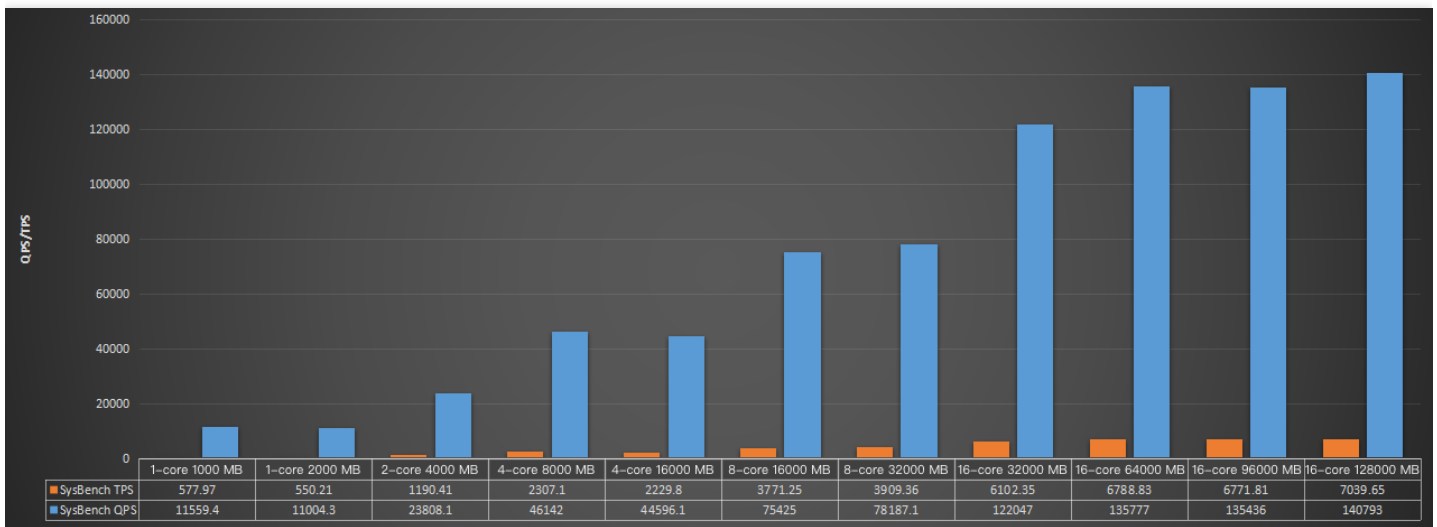
MySQL 5.6 테스트 결과

최종 업데이트 날짜 : 2022-03-31 15:08:06

본 문에서는 TencentDB for MySQL 5.6 범용 인스턴스용 성능 테스트 결과를 소개합니다.

시나리오1: 전체 캐시

전체 캐시는 모든 데이터만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하기 위해 디스크를 읽고 쓸 필요가 없음을 의미합니다.

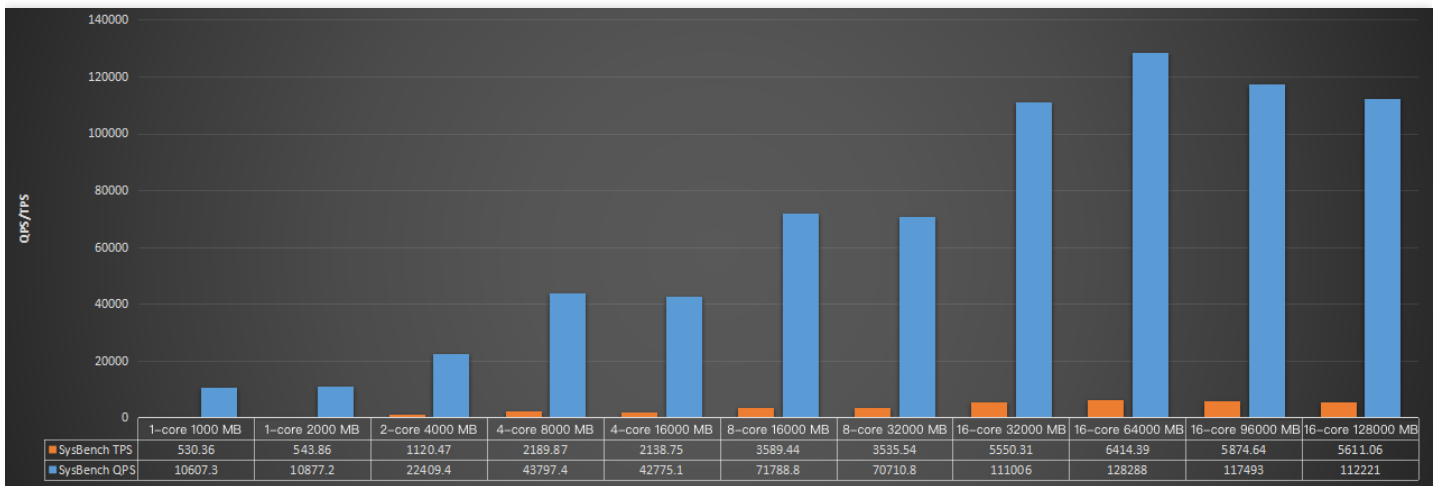


CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	25000	150	577.97	11559.4	13.84
1	2000	8	25000	150	550.21	11004.3	14.54
2	4000	16	25000	150	1190.41	23808.1	13.44
4	8000	32	25000	150	2307.1	46142	13.87
4	16000	32	25000	150	2229.8	44596.1	14.35
8	16000	64	25000	150	3771.25	75425	16.97
8	32000	64	25000	150	3909.36	78187.1	16.37
16	32000	128	25000	150	6102.35	122047	20.97

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
16	64000	128	25000	150	6788.83	135777	18.85
16	96000	128	25000	150	6771.81	135436	18.9
16	128000	128	25000	150	7039.65	140793	18.18

시나리오2: 디스크 IO 유형

디스크 IO 유형 시나리오는 데이터의 일부만 캐시에 넣을 수 있으며 쿼리 프로세스 중에 캐시를 업데이트하려면 디스크를 읽기/쓰기 해야 합니다.



CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
1	1000	8	800000	6	530.36	10607.3	15.08
1	2000	8	800000	12	543.86	10877.2	14.71
2	4000	16	800000	24	1120.47	22409.4	14.28
4	8000	32	800000	48	2189.87	43797.4	14.61
4	16000	32	6000000	13	2138.75	42775.1	14.96
8	16000	64	6000000	13	3589.44	71788.8	17.83
8	32000	64	6000000	25	3535.54	70710.8	18.1

CPU (core)	메모리 (MB)	동시성	단일 테이블의 데이터 양	총 테이블 수	SysBench TPS	SysBench QPS	avg_lat
16	32000	128	6000000	25	5550.31	111006	23.06
16	64000	128	6000000	49	6414.39	128288	19.95
16	96000	128	6000000	74	5874.64	117493	21.78
16	128000	128	6000000	98	5611.06	112221	22.81

네트워크 아키텍처 성능 비교

최종 업데이트 날짜: : 2023-03-13 12:01:58

TencentDB for MySQL은 더 높은 성능과 더 짧은 대기 시간을 위해 데이터베이스 인스턴스의 네트워크 아키텍처를 업그레이드했습니다. 본문은 이전 버전과 새 버전의 성능 테스트에 대해 설명합니다.

설명 :

- **2022년 11월 9일부터 새로 구입한 인스턴스에 새로운 네트워크 아키텍처가 적용되어 더 짧은 지연 시간과 더 높은 성능을 제공합니다.**
- **2023년 1월 21일 모든 기존 데이터베이스 인스턴스가 새로운 네트워크 아키텍처로 전환됩니다.** 데이터베이스에서 일시적으로 연결이 끊어질 수 있습니다. 비즈니스에 재연결 메커니즘이 있는지 확인하십시오.
- 클라우드 디스크 버전의 단일 노드 인스턴스는 이미 최적의 네트워크 아키텍처에 있으므로 세부 정보 페이지에 네트워크 아키텍처가 새로운지 여부가 표시되지 않습니다.
- 기존 네트워크에서는 새로운 네트워크 아키텍처를 사용할 수 없습니다. 이를 사용하려면 [네트워크 변경](#)의 안내에 따라 VPC로 전환하고 네트워크 아키텍처가 업그레이드될 때까지 기다립니다.
- 자세한 내용은 [네트워크 아키텍처 업그레이드](#)를 참고하십시오.

테스트 환경

- 리전/AZ: 베이징 - 베이징 6존.
- 클라이언트 사양: S5.2XLARGE16(8개의 CPU 코어 및 16GB 메모리).
- 클라이언트 운영 체제: TencentOS Server 3.2.
- 네트워크: CVM과 TencentDB for MySQL 인스턴스는 모두 동일한 VPC 서브넷에 있습니다.
- 스토리지 유형: 로컬 SSD 디스크.
- 인스턴스 사양: 일반(4개의 CPU 코어 및 16GB 메모리).
- 매개변수 템플릿: 고성능 템플릿.
- 복제 방식: 비동기 복제.

테스트 툴

SysBench는 집약적인 부하에서 데이터베이스를 실행하는 시스템에 중요한 OS 매개변수를 평가하기 위한 모듈식, 크로스 플랫폼 및 멀티 스레드 벤치마크 툴입니다. 이 벤치마크 제품군의 아이디어는 복잡한 데이터베이스 벤치마크를 설정하지 않거나 데이터베이스를 전혀 설치하지 않고도 시스템 성능에 대한 인상을 신속하게 얻는 것입니다. 이 스트레스 테스트는 SysBench 1.0.20을 사용합니다.

테스트 시나리오

이 스트레스 테스트에는 쓰기 전용, 읽기 전용 및 읽기-쓰기 시나리오가 포함되며 각각 2 - 3000개의 스레드가 테스트됩니다. QPS는 성능 메트릭으로 사용됩니다.

테스트 방법

1단계: 데이터 준비

다음 명령을 실행합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX
--mysql-user=XXX --mysql-password=XXX --mysql-db=sbtest --table_size=10000000
--tables=10 --events=0 --time=300 --threads={2~3000} oltp_read_write prepare
```

2단계: workload 실행

쓰기 전용, 읽기 전용 및 읽기-쓰기 시나리오에서 **workload**를 실행합니다. 구성이 올바른지 확인하십시오.

- OLTP 쓰기 전용 시나리오

다음 명령을 실행합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX
--mysql-user=XXX --mysql-password=XXX --mysql-db=sbtest --table_size=10000000
--tables=10 --events=0 --time=300 --threads={2~3000} --percentile=95 --report-i
nterval=1 oltp_write_only
run
```

- OLTP 읽기 전용 시나리오

다음 명령을 실행합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX
--mysql-user=XXX --mysql-password=XXX --mysql-db=sbtest --table_size=10000000
--tables=10 --events=0 --time=300 --threads={2~3000} --percentile=95 --skip-trx
=1 --report-interval=1
oltp_read_only
run
```

- OLTP 읽기-쓰기 시나리오

다음 명령을 실행합니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX
--mysql-user=XXX --mysql-password=XXX --mysql-db=sbtest --table_size=10000000
--tables=10 --events=0 --time=300 --threads={2~3000} --percentile=95 --report-i
nterval=1 oltp_read_write
run
```

3단계: 데이터 지우기

테스트가 실행된 후 다음 명령을 실행하여 데이터를 지웁니다.

```
sysbench --db-driver=mysql --mysql-host=XXX --mysql-port=XXX --mysql-user=XXX --m
ysql-password=XXX
--mysql-db=sbtest --table_size=25000 --tables=250 --events=0 --time=600 --threads
=XXX --percentile=95 oltp_read_write cleanup
```

테스트 메트릭

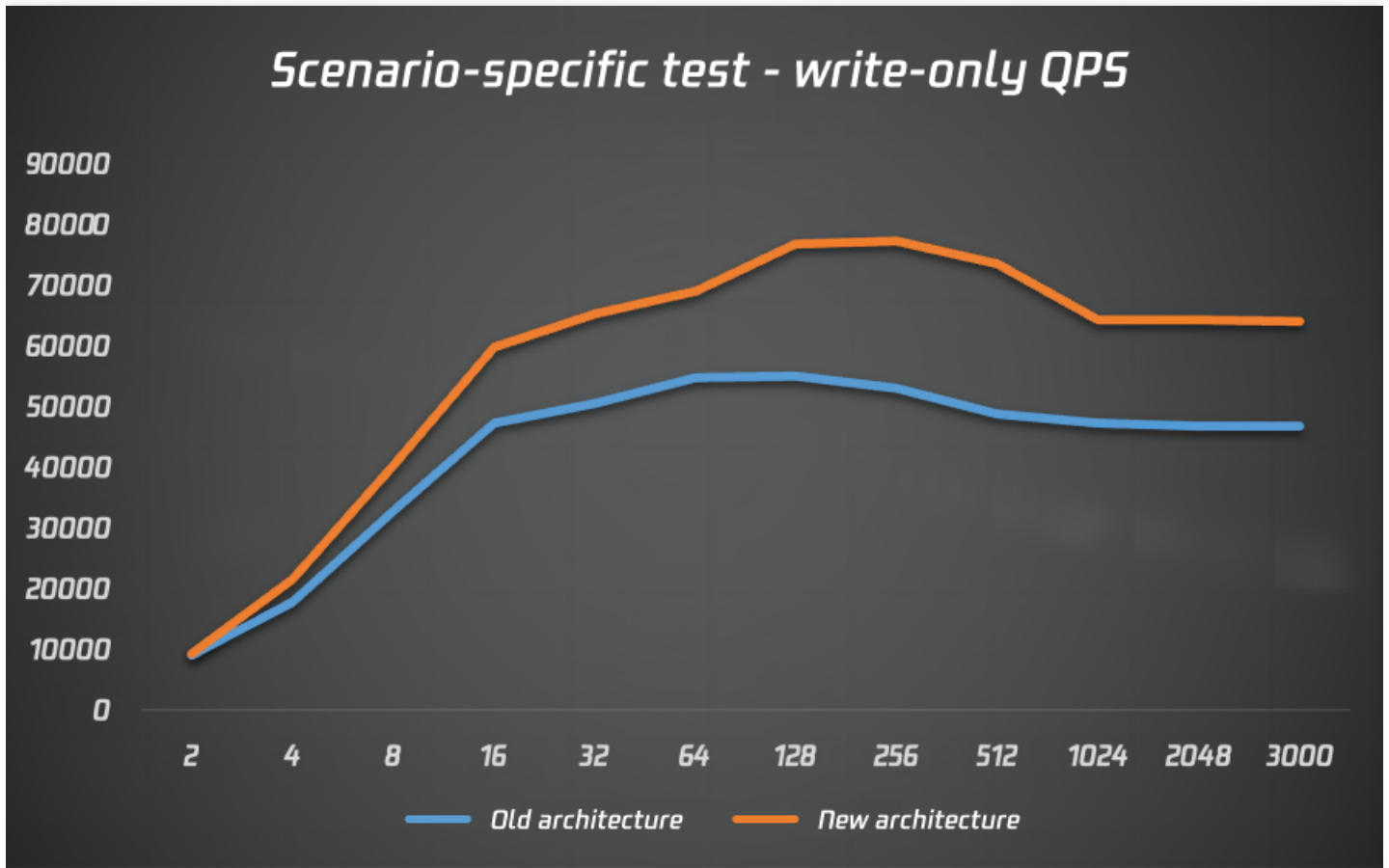
테스트 메트릭은 초당 쿼리 수(QPS)입니다.

테스트 결과

쓰기 전용 시나리오의 테스트 결과

쓰기 전용 시나리오에서 TencentDB for MySQL의 새로운 아키텍처는 스레드 수가 증가함에 따라 항상 기존 아키텍처보다 성능이 우수합니다. QPS는 256개 스레드에서 정점에 도달하고 이전 아키텍처의 512개 스레드보다 20% 더 높

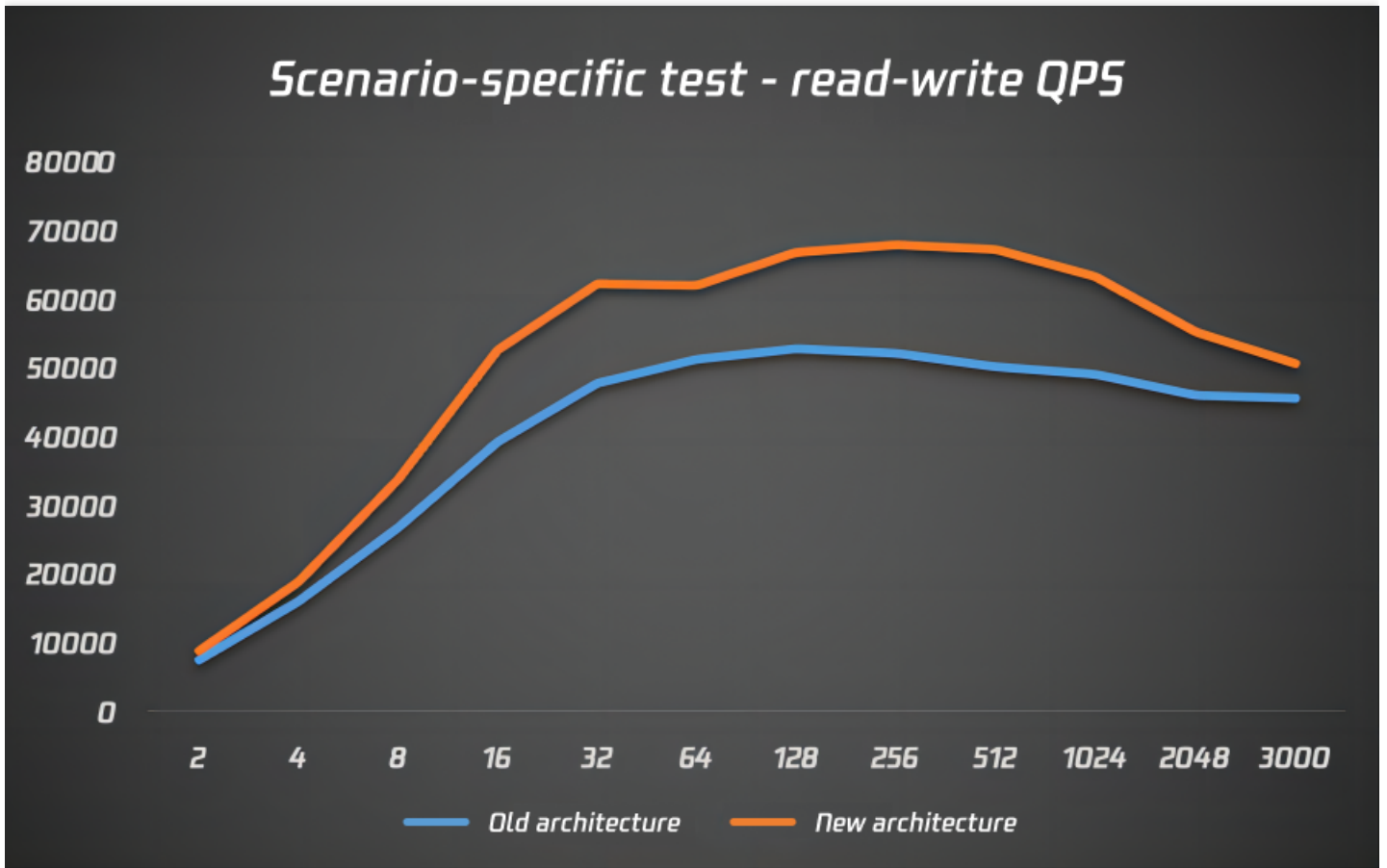
습니다.



읽기 전용 시나리오의 테스트 결과

읽기 전용 시나리오에서 TencentDB for MySQL의 새로운 아키텍처는 스레드 양이 작은 수에서 증가함에 따라 선형 QPS가 크게 증가하는 것을 확인했으며, QPS는 기존 아키텍처의 16 스레드보다 22% 더 높고 64 스레드 이상에서 꾸

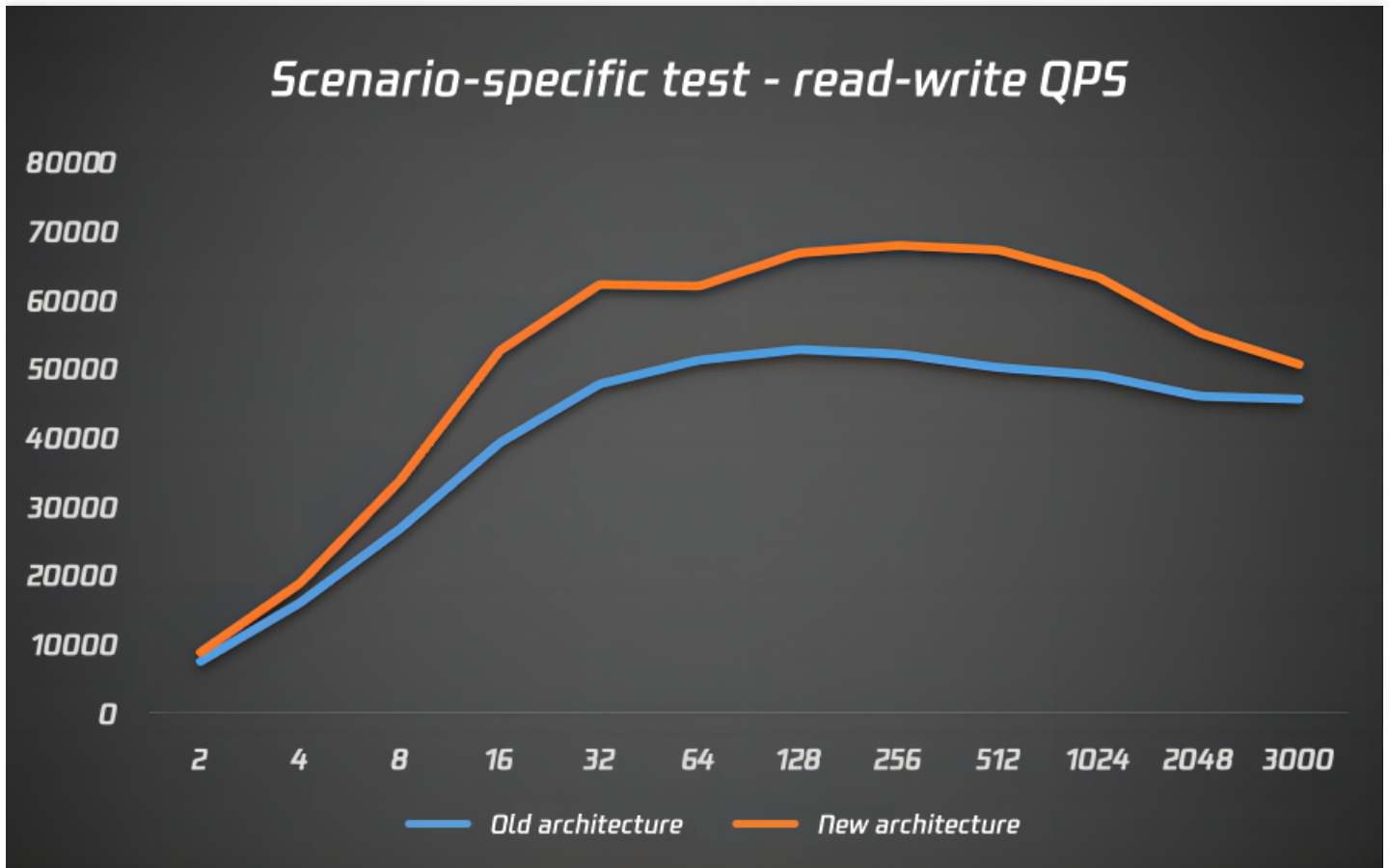
준히 증가하고 있습니다.



읽기-쓰기 시나리오의 테스트 결과

읽기-쓰기 시나리오에서 TencentDB for MySQL의 새로운 아키텍처는 스레드 수가 적은 수에서 증가함에 따라 QPS가 크게 증가하는 것을 확인합니다. QPS는 512 스레드에서 정점(이전 아키텍처보다 18% 더 높음)에 도달한 후 꾸준

히 감소합니다.



결론

설명 :

상기 성능 테스트 결과는 참고용입니다.

상기 시나리오 테스트는 TencentDB for MySQL의 새로운 네트워크 아키텍처가 이전 아키텍처보다 훨씬 뛰어난 성능을 보인다는 것을 보여줍니다. 2~3000 스레드 범위에서 QPS가 평균 20% 이상 향상됩니다.

보안 백서

개요

최종 업데이트 날짜: : 2022-10-20 17:15:50

TencentDB for MySQL은 사용자가 클라우드에서 MySQL 데이터베이스를 간편하게 배포 및 사용하도록 지원합니다. TencentDB for MySQL을 통해 몇 분 만에 확장 가능한 MySQL 데이터베이스 인스턴스를 더 저렴한 비용으로 배포하고 서버 다운 없이 하드웨어 용량을 조정할 수 있습니다. TencentDB for MySQL은 백업/롤백/모니터링/빠른 확장/데이터 전송 등 데이터베이스 유지보수에 대한 통합 솔루션을 제공하여 IT 유지보수 작업을 간소화하고 사용자가 비즈니스 확장에 집중할 수 있도록 돕습니다.

TencentDB for MySQL은 다양한 보안 강화 기능을 제공하여 사용자 데이터의 신뢰성과 보안성을 보장합니다. MySQL 데이터베이스의 보다 강력한 보안성을 위해, 필요에 따라 다음과 같은 보안 기능 사용을 권장합니다.

보안 기능	기능 항목
데이터 스토리지 보안	<ul style="list-style-type: none"> • 자동 백업 • 아카이브 백업 보관 • TDE • 데이터 보안
보안 감사	<ul style="list-style-type: none"> • 컴플라이언스 감사 • 보안 거버넌스
액세스 제어	<ul style="list-style-type: none"> • 데이터베이스 계정 관리 • 액세스 관리 • 비밀번호 복잡성
데이터 통신 보안	<ul style="list-style-type: none"> • VPC • 보안 그룹 • SSL 암호화
데이터 재해 복구	<ul style="list-style-type: none"> • 리전 내 재해 복구 • 리전 간 재해 복구

데이터 스토리지 보안

최종 업데이트 날짜: : 2023-07-06 16:40:15

TencentDB for MySQL는 데이터 스토리지의 높은 보안성을 보장합니다. 저장 과정 중 데이터의 기밀성과 완전성을 보장합니다.

자동 백업

TencentDB for MySQL은 자동 백업과 수동 백업을 모두 지원하여 데이터의 완전성과 신뢰성을 보장하는 데이터 복원성을 보장합니다. MySQL은 기본적으로 데이터 백업과 로그 백업 기능을 제공하며, 자동 백업 주기는 한 주에 2회 이하로 설정할 수 없습니다. 기타 백업이 필요할 경우, 콘솔 혹은 API를 통해 언제든지 수동 백업을 실행할 수 있습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [자동 백업 설정](#)을 참고하십시오.

아카이브 백업 보관

TencentDB for MySQL을 사용하면 아카이브 백업 보관을 활성화할 수 있습니다. 필요에 따라 백업 파일의 보관 기간을 유연하게 구성할 수 있습니다. 이 기간은 기본적으로 7일이며 최대 1830일까지 가능합니다. 보관 기간을 초과한 백업 파일은 자동으로 삭제됩니다. 데이터 보안을 위해 일주일에 두 번 이상 데이터를 백업하는 것이 좋습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [정기 백업 설정 활성화](#)를 참고하십시오.

Transparent Data Encryption

TencentDB for MySQL는 Tencent Cloud 데이터베이스 팀에서 개발한 Transparent Data Encryption(TDE) 기능을 지원합니다. 투명한 암호화는 데이터 암호화 및 암호 해독이 사용자에게 감지되지 않음을 의미합니다. 암호화된 테이블을 생성할 때 암호화 키를 지정할 필요가 없으며 디스크에 쓰는 동안 데이터는 암호화되고 디스크에서 읽는 동안 해독됩니다.

TDE는 국제적으로 통용되는 AES 알고리즘과 256비트 암호화 키를 사용하며, Tencent Cloud [Key Management Service](#)가 관리합니다. KMS 서비스에 액세스하려면 액세스 권한이 있어야 하며 KMS 콘솔에서 키를 교체하여 시스템 보안을 더욱 강화할 수 있습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [TDE 활성화](#)를 참고하십시오.

액세스 제어

최종 업데이트 날짜: : 2022-10-20 17:15:51

TencentDB for MySQL는 액세스 제어 기능을 제공합니다. 사용자 권한을 정의 및 확인하고, 데이터베이스 리소스에 대한 사용자 액세스를 규제하고, 데이터베이스 리소스 권한을 관리함으로써 권한이 있는 사용자만 권한 범위 내에서 또는 보안 수준에서 데이터베이스 객체에 액세스할 수 있도록 할 수 있습니다.

데이터베이스 계정 관리

TencentDB for MySQL 콘솔 또는 API를 통해 데이터베이스 계정을 생성할 수 있습니다. 이러한 계정에 다양한 레벨의 관리 권한을 부여할 수도 있습니다. 데이터 보안을 위해 최소 권한 원칙에 따라 계정을 승인하는 것이 좋습니다.

자세한 내용은 [계정 생성](#)을 참고하십시오.

액세스 관리

액세스 관리(Cloud Access Management, CAM)를 사용하면 Tencent Cloud 리소스에 대한 액세스 권한을 안전하게 관리하고 제어할 수 있습니다. CAM을 사용하면 사용자(그룹)를 생성, 관리 및 종료할 수 있으며 권한 분리를 구현하는 ID 및 정책 관리를 통해 지정된 사용자가 사용할 수 있는 Tencent Cloud 리소스를 제어할 수 있습니다.

자세한 내용은 [CAM Overview](#)를 참고하십시오.

비밀번호 복잡성

비밀번호는 데이터베이스 보안을 보호하는 가장 중요한 수단입니다. 더 많은 데이터 보안 규정이 도입됨에 따라 데이터베이스 비밀번호 강도에 대한 요구 사항이 높아졌습니다. TencentDB for MySQL은 사용자 지정 비밀번호 복잡성 기능을 지원하여 데이터베이스 보안을 보호하고 해당 규정의 컴플라이언스 요구 사항을 충족합니다.

콘솔에서 사용자 정의 암호 복잡성 기능을 구성하여 비밀번호와 관련된 모든 콘솔 및 데이터베이스 작업에 대한 비밀번호 강도를 설정할 수 있습니다. 이렇게 하면 비밀번호를 보호하고 비밀번호 유출과 같은 보안 위험을 방지할 수 있습니다. 이 기능은 다음 구성 항목을 제공합니다.

- 소문자 및 대문자의 최소 글자 수.
- 숫자의 최소 글자 수.
- 특수 문자의 최소 글자 수.
- 비밀번호 최소 글자 수.

자세한 내용은 [Setting Password Complexity](#)를 참고하십시오.

데이터 통신 보안

최종 업데이트 날짜: : 2022-10-20 17:15:51

TencentDB for MySQL은 통신 중 데이터 통신 보안 기능을 제공하여 데이터의 기밀성과 완전성을 보장합니다.

VPC

TencentDB for MySQL은 더 높은 수준의 네트워크 격리 및 제어를 달성하기 위해 Virtual Private Cloud(VPC) 사용을 지원합니다. VPC는 Tencent Cloud에서 논리적으로 격리된 네트워크 공간입니다. VPC에서 IP 범위, IP 주소 및 라우팅 정책을 사용자 지정하여 리소스 수준에서 네트워크 격리를 구현할 수 있습니다.

VPC에 배포된 MySQL용 TencentDB 인스턴스는 기본적으로 동일한 VPC의 CVM 인스턴스로만 액세스할 수 있습니다. CVM과 MySQL 인스턴스가 서로 다른 VPC에 있는 경우 공중망 액세스를 신청한 후에 통신할 수 있습니다. 네트워크 보안을 위해 공중망을 통해 데이터베이스에 액세스하지 않는 것이 좋습니다. 공중망을 통해 액세스해야 하는 경우 적절한 보안 그룹을 구성하여 클라이언트에 대한 액세스 제어를 구현하십시오.

이 기능을 사용하는 방법에 대한 자세한 내용은 [TencentDB for MySQL을 위한 VPC 생성](#)을 참고하십시오.

보안 그룹

TencentDB for MySQL은 하나 이상의 TencentDB 인스턴스에 대한 네트워크 액세스 제어를 설정하는 데 사용할 수 있는 네트워크 보안 격리를 위한 중요한 수단으로 보안 그룹을 지원합니다. 한 리전에서 동일한 네트워크 보안 격리 요구 사항이 있는 인스턴스를 동일한 보안 그룹(논리적 그룹)에 넣을 수 있습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [CDB 보안 그룹 관리](#)를 참고하십시오.

SSL 암호화

TencentDB for MySQL은 사용자 클라이언트에서 TencentDB 서버로의 연결을 인증하는 프로세스인 SSL(Secure Sockets Layer) 인증을 지원합니다. SSL 암호화가 활성화되면 CA 인증서를 가져와 서버에 업로드할 수 있습니다. 그런 다음 클라이언트가 데이터베이스에 액세스하면 SSL 프로토콜이 활성화되어 클라이언트와 서버 간에 SSL 보안 채널이 설정됩니다. 이는 암호화된 데이터 전송을 구현하고 전송 중 데이터 가로채기, 변조 및 도청을 방지하여 궁극적으로 데이터 보안을 보장합니다.

SSL 암호화는 데이터 자체를 보호하지 않습니다. 대신 데이터베이스와 서버 간의 트래픽을 보호합니다. 전송 레이어에서 네트워크 연결을 암호화하면 통신 데이터의 보안과 완전성을 향상시킬 수 있지만 네트워크 연결의 응답 시간이 늘어납니다.

이 기능의 자세한 사용 방법은 [SSL 암호화 설정](#)을 참고하십시오.

데이터 재해 복구

최종 업데이트 날짜: : 2022-10-20 17:15:51

TencentDB for MySQL은 데이터 안정성을 향상시키면서 저렴한 비용으로 지속적인 서비스를 제공할 수 있도록 가용존 및 리전 간 데이터 재해 복구 솔루션을 제공합니다. 컴플라이언스가 필요한 시나리오에도 적용할 수 있습니다.

리전 내 재해 복구

[2노드](https://www.tencentcloud.com/document/product/236/39783) 및 [\[3노드\]](https://www.tencentcloud.com/document/product/236/39783)(<https://www.tencentcloud.com/document/product/236/39783>) TencentDB for MySQL 인스턴스를 생성하여 멀티 AZ(가용존) 배포를 지원할 수 있습니다. 멀티 AZ 인스턴스의 물리적 서버는 동일한 리전의 다른 AZ에 배포됩니다. AZ에 장애가 발생하면 비즈니스 트래픽이 다른 AZ로 신속하게 전환되므로 비즈니스에 투명하며 리전 내 재해 복구를 위해 애플리케이션 레이어를 변경할 필요가 없습니다.

설명 :

멀티 가용존 인스턴스가 서로 다른 가용존에 있으면, 네트워크 동기화가 2 - 3ms 가량 딜레이 될 수 있습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [고가용성\(멀티 가용존\)](#)을 참고하십시오.

리전 간 재해 복구

TencentDB for MySQL의 리전 내 재해 복구 기능은 동일한 리전의 다른 AZ로 제한됩니다. 가용성을 더욱 향상시키기 위해 TencentDB for MySQL는 리전 간 데이터 재해 복구도 지원합니다.

데이터 전송을 통해 리전 A의 MySQL용 TencentDB 인스턴스의 데이터를 리전 B의 다른 인스턴스(재해 복구 인스턴스)로 비동기식으로 복제할 수 있습니다. 재해 복구 인스턴스는 독립적인 연결 주소, 계정 및 권한이 있습니다. A 리전에 단기 내 복구가 불가능한 중대 장애가 발생한 경우, 언제든지 재해 복구로 전환할 수 있습니다. 특히 애플리케이션에서 데이터베이스 연결 구성을 수정하기만 하면 재해 복구 인스턴스에 애플리케이션 요청을 신속하게 전달할 수 있으므로 금융급 데이터베이스 가용성을 제공할 수 있습니다.

이 기능을 사용하는 방법에 대한 자세한 내용은 [재해 복구 인스턴스](#)를 참고하십시오.