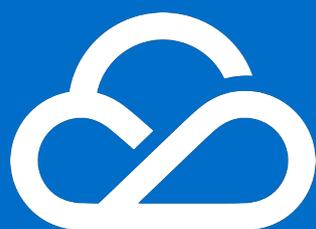


云数据库TDSQL

用户手册

产品文档



腾讯云

【版权声明】

©2015-2016 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

文档声明.....	2
兼容性说明.....	4
功能限制	7
读写分离	8
实例接入	9
数据迁入.....	13
数据迁出	19
解压备份和日志文件	20
CLI参考	22
恢复备份	23
数据库审计	29
数据库审计已支持语法.....	36
实例类型.....	38

兼容性说明

TDSQL内核采用MariaDB 10.0修改，兼容性与MariaDB 10.0基本相同，这里明确如下：

TDSQL与MySQL兼容性

2015年12月31日更新

- 数据文件和表定义文件是二进制兼容的；
- 所有的客户端API和协议都是兼容的；
- 所有的文件名、二进制文件、路径、端口号等都是相同的；
- 所有的连接器，包括PHP、Perl、Python、Java、.NET、Ruby、MySQL的连接器在MariaDB上都可以正常使用的，不需要进行任何改动。
- 可以使用MySQL客户端链接到TDSQL上。

TDSQL和MySQL5.6的不兼容性

2015年12月31日更新

- TDSQL的GTID和MySQL 5.6的GTID不能兼容，也就是说MySQL不能作为TDSQL的从库；
- 如果仅仅给出选项的前缀部分，例如使用--big-table取代--big-tables，MySQL将会给出警告，而TDSQL将会正常工作。也就是说，只要给出的前缀部分能够唯一地标识该选项即可。
- 为了使CREATE TABLE ...
SELECT命令在基于行模式复制和基于命令模式复制的情况下都能正常工作，TDSQL中的CREATE TABLE ... SELECT命令在从库上将会被转化为CREATE OR REPLACE命令执行。这样的好处是即使从库在执行CREATE TABLE ... SELECT命令时宕机了，仍然能够正常工作。

2016年1月12日更新

- TDSQL的innodb_page_size默认设置为4KB，MySQL默认设置为16KB，会导致单个索引不能超过768字节

2016年1月22日更新

- 在JDBC应用程序下，如果使用mysql-connector-java.jar，在执行如下子查询SQL时可能报错：

```
select * from (select col_a , col_b , from tbl_a where col_b=1) tx
where tx.col_a = '01';
Table db.tx' doesn't exist
```

- 使用mariadb-java-client-1.2.x.jar时，在执行如下子查询SQL时可能出现异常

```
select col_a as a from tbl;
a
col_a
```

- 使用mariadb-java-client-1.3.x.jar时，会遇到如果设置了autocommit属性（global或者session），那么执行commit方法就会报错。
- 经测试表明，在Hibernate、entityframework框架中，如果设置参数useAffectedRows=false，并要求update语句返回matched rows时，由于Mariadb忽略了客户端在建立连接阶段推送的CLIENT_FOUND_ROWS字段，会导致useAffectedRows=false参数的设置不生效，使得update实际返回affected rows。

```
: update tal_a set col_a = 'value';
col_a'value'affected rows=0matched rows=1
```


功能限制

功能限制

1.不能更改mysql、information_schema、performance_schema和sysdb库中的任何数据

2.无法直接通过SQL语句进行帐号和赋权操作，只能通过管理控制台进行

- 支持常见的19种权限，少量不常见权限不支持，具体支持的权限列表如下：

- SELECT、INSERT、UPDATE、DELETE、CREATE、DROP、REFERENCES、INDEX、ALTER
- CREATE TEMPORARY TABLES、LOCK TABLES、EXECUTE、CREATE VIEW、SHOW VIEW
- CREATE ROUTINE、ALTER ROUTINE、EVENT、TRIGGER、SHOW DATABASES

3.不提供超级管理员帐号

4.采用InnoDB存储引擎，暂不开放其他存储引擎

读写分离

1. 背景说明

云数据库（TDSQL）默认支持读写分离能力，标准版一个数据库节点组（SET）基本配置为一主二从，两从机默认可提供读能力。

为保证不影响集群稳定性，TDSQL的从机读并不需要用户直连从机，而是由网关集群（TProxy）自动分配到低负载从机上。

实现方法

在每条需要从机“读”的语句前，增加/slave/字段，并且mysql后面要增加-c 参数来解析注释mysql -c -e "/*slave*/sql"，即可自动将读请求分配到从机，示例如下：

```
//□□□//  
select * from emp order by sal□deptno desc□  
//□□□//  
/*slave*/ select * from emp order by sal□deptno desc□
```

注意事项：

- 仅支持从机读（select），不支持其他操作，非select语句将失败；
- mysql后面要增加-c 参数来解析注释
- /*slave*/必须为小写，语句前后无空格；
- 从机出现异常而影响到MAR（强同步）机制时，从机读操作将自动切换回主机。

实例接入

您可以通过“创建账号”、“获取实例地址”两步轻松连接CDB for TDSQL实例。

1.创建账号

1.1. 进入账号管理页面，选择账号管理功能

在您正式开始使用TDSQL服务之前，您需要进入TDSQL实例的管理页面，创建用于登陆TDSQL的数据库账号，并设置相应的访问权限。



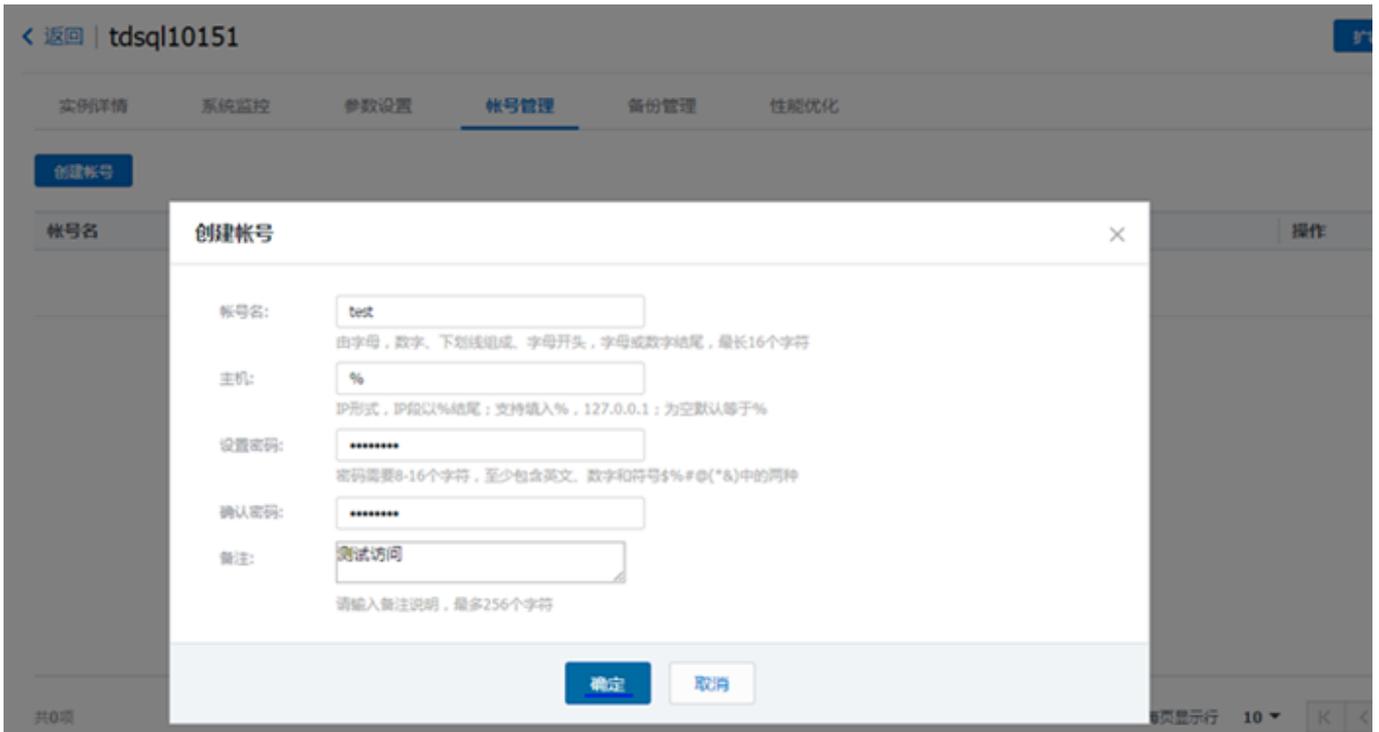
您可以在管理页面看到“帐号管理”，如下图所示：



1.2. 创建账号

这里的账号由账号名、主机、密码三部分组成；

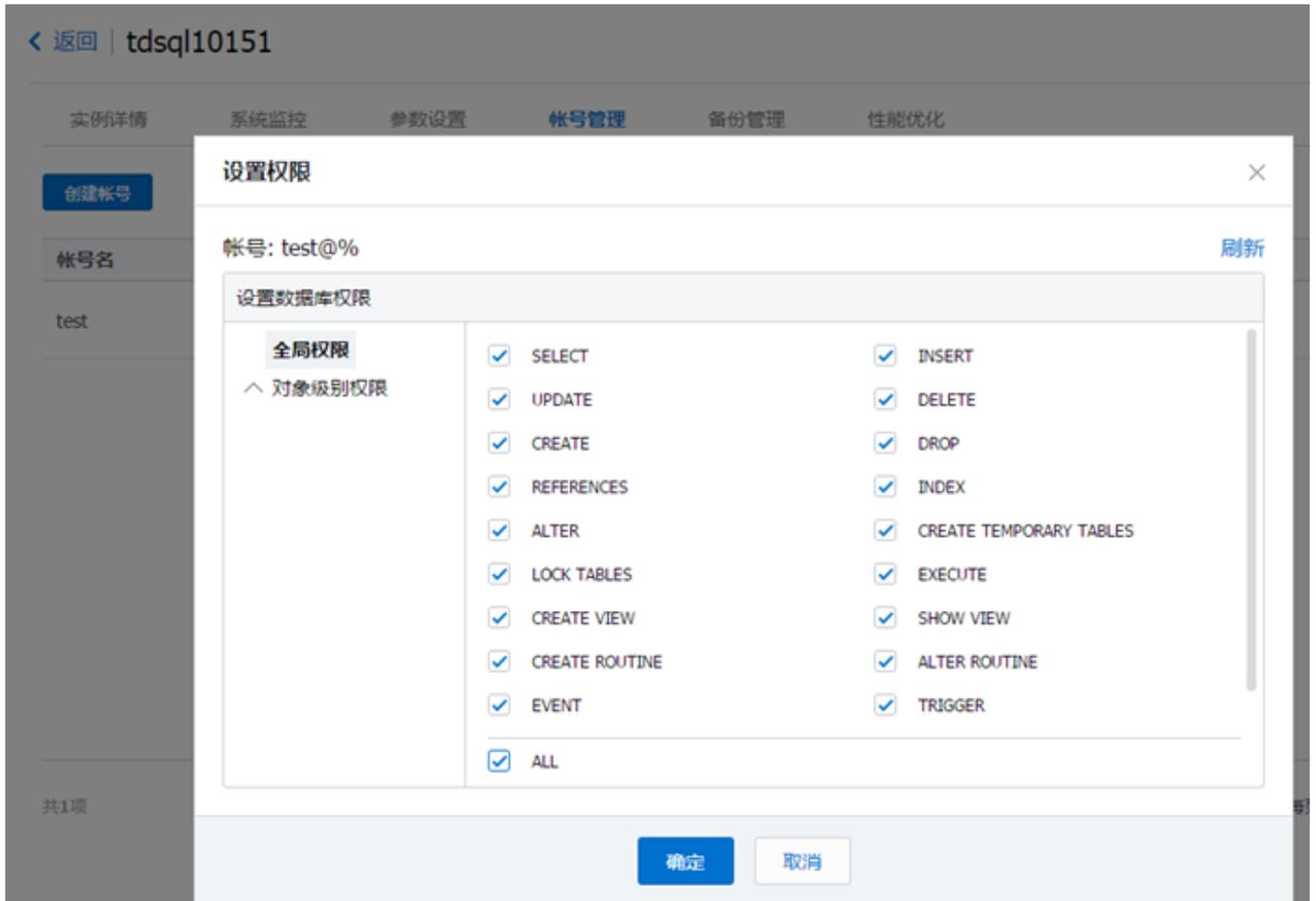
- 账号名：由字母、数字、下划线组成，字母开头，字母或数字结尾，最长16个字符组成；
- 主机
 - ：支持IP、IP段、%三种形式；%代表结尾符，例如我们要支持10.10.10.1~10.10.10.254的所有主机IP，可以输入10.10.10.%；
- 密码
 - ：密码为8~16位的任意字符，至少包括英文、数字和符号等；根据国家信息安全相关规定，腾讯云建议但不强制用户每3个月修改一次密码。



注意：同一个帐号名不同的主机IP需要可能独立设置权限，您可以选择账号管理中“克隆”功能，克隆相似的帐号和权限设置。

1.3. 设置权限

TDSQL的权限为对象级权限，权限包括MySQL常见的19种权限；也可以为表、视图、函数、触发器等对象设置权限。



注意：未创建数据库时，无法设置对象级别权限。

2. 获取内网地址

在实例详情页面，直接获取实例内网地址。说明一下，这里的内网地址是VIP，是通过接入网关集群统一访问数据库实例，而非直接连接到数据库实例物理机上，因此是否主机发送故障或主备切换，内网IP都不会变化。

[< 返回](#) | tdsq110151

[实例详情](#)

[系统监控](#)

[参数设置](#)

[帐号管理](#)

[备份管理](#)

[性能优化](#)

基本信息

实例ID	427b	8e87
运行状态	运行中	
内网地址	10.66.89.1 ~ 10.66.89.36	
实例类型	小规格	
内存	6 GB	
总容量	100GB 扩容	
binlog容量	25 GB	
地域	华南区-广州	
所在网络	基础网络	
所属项目	默认项目 转至其他项目	
到期时间	2015-12-30 19:14:36	
创建时间	2015-11-30 19:14:36	

您可以在和TDSQL所属网络相同的云服务器上，直接连接实例的内网地址访问TDSQL服务。（注意：您可能需要先安装mysql客户端）

```
VM_46_138_suse:~ # mysql -utest -pqqqqqq -h10.66.89.109 -P3306
Warning: Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1209203210
Server version: 5.0.81-log Source distribution

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sysdb |
| test |
+-----+
5 rows in set (0.01 sec)
```

数据迁入

1.通过腾讯云迁移工具迁入

您可以通过腾讯云数据库迁移工具将业务数据迁移到CDB for

TDSQL中，迁移工具可以把同地域下不同类型的源数据库的数据迁移到TDSQL中，目前支持的源库类型有：

1. 源数据库来自在腾讯云基础网络中云服务自建的MySQL实例（即将支持[VPC网络](#)）；
2. 源数据库在腾讯云的CDB for MySQL实例；
3. 源数据库在另外一个CDB for TDSQL实例；

迁移步骤如下：

1.1. 进入迁移工具页面



The screenshot shows the Tencent Cloud console interface for TDSQL instances. The top navigation bar includes '腾讯云', '总览', '产品服务', '云数据库', '云存储Redis', '对象存储服务', 'TDSQL', 'CDN', and '+ 自定义'. The main content area is titled 'TDSQL' and shows a list of instances. A blue warning banner at the top states: '为了您的数据安全，请不要手动修改或删除系统默认的mysql库中任意表的数据!' (For your data security, please do not manually modify or delete data in any table of the system default mysql database!). Below this is a '+新建' (New) button and a table of instances:

<input type="checkbox"/> 实例名称	运行状态	规格	所属网络
<input type="checkbox"/> tdsql10151	运行中	6G内存 100G存储空间	基础网络
<input type="checkbox"/> tdsql10135	运行中	6G内存 100G存储空间	基础网络

In the left sidebar, the '数据库列表' (Database List) menu is expanded, and '数据库迁移' (Database Migration) is highlighted with a red rectangular box.

1.2. 创建迁移任务

云数据库-迁移工具

华南地区 (广州) ▾

[+ 创建迁移任务](#) [启动](#) [中止](#) [完成](#) [删除](#) [筛选 ▾](#) 排序: [创建时间 ▾](#) [降序 ▾](#) [任务名 ▾](#) [Q](#)

<input type="checkbox"/> 全选	状态	创建时间	操作
<input type="checkbox"/> 名称: 迁移任务	状态: 迁移成功	2015-12-11 19:33:53	查看任务 修改 删除 校验
2015-12-11 19:52:28 开始			
<div style="background-color: #27ae60; color: white; padding: 2px;">数据迁移 100%</div>			
<input type="checkbox"/> 名称: testabc	状态: 初始化	2015-12-09 19:38:26	查看任务 修改 删除 校验 ⊙
未开始, 待启动			
未完成			

点击创建迁移任务，输入任务名称、源库和目标TDSQL的信息。

 任务初始化

任务设置

任务名称 *

数据库迁移



定时执行



定时执行意味着该任务从指定时间开始排队

源库信息

源库类型 *

自建MySQL



CVM实例ID *

ins-f11...



IP *

10.232.10.112



端口 *

3306



账号 *

root



密码 *

.....



目标库信息

目标库类型 *

CDB for TDSQL

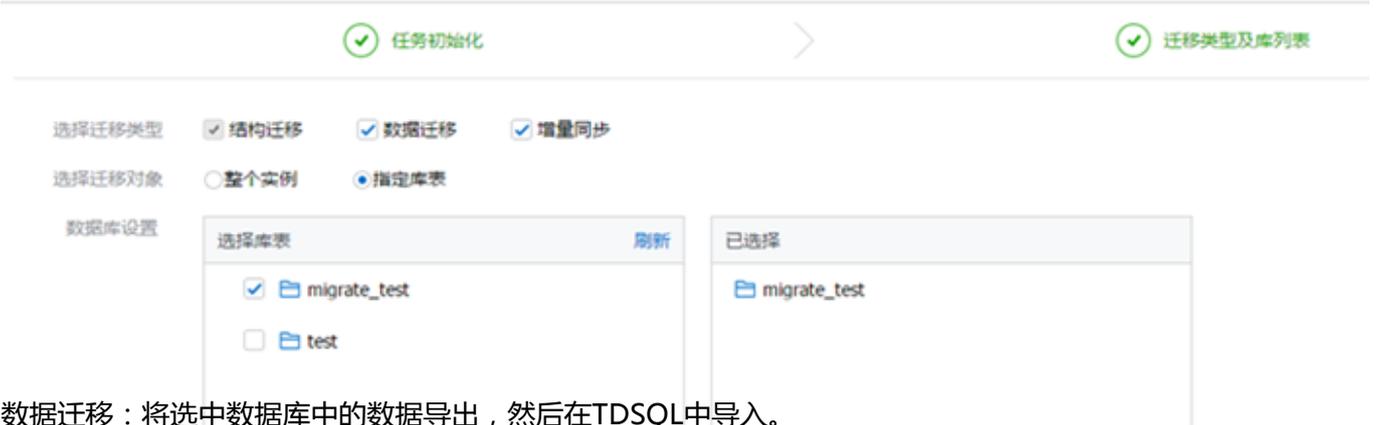


实例ID *

427b70f7-11e4-488f-8c85-7c31c63c8e8



然后选择要迁移的数据库，创建并检查迁移任务信息。



数据迁移：将选中数据库中的数据导出，然后在TDSQL中导入。

增量同步：在进行数据导出导入后，设置TDSQL为源库的备库，进行主备增量同步。

1.3. 校验迁移任务信息

在创建完迁移任务后，您需要对迁移任务信息进行校验，只有所有校验项通过后才能启动迁移任务。任务校验存在3种状态：

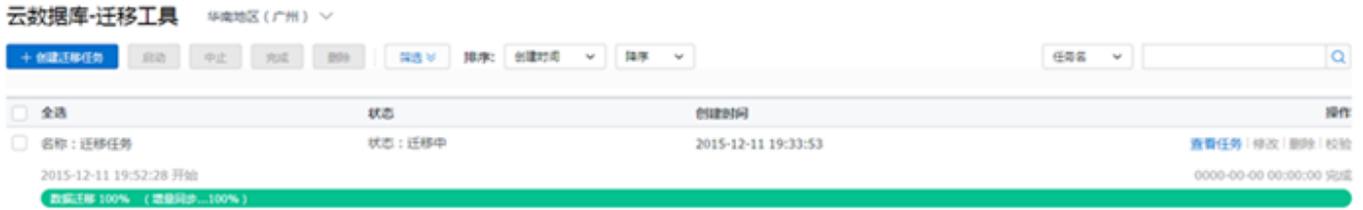
- 通过：表示校验完全通过
- 警告：表示校验不通过，迁移过程中或迁移后可能影响数据库正常运行但不影响迁移任务的执行。
- 失败：表示校验不通过，无法进行迁移。如果校验失败，请根据出错的校验项，检查并修改迁移任务信息，然后重试校验。失败原因请参考：“校验失败说明”



1.4. 启动迁移

在校验通过后，您可以启动迁移任务，如果您设定了迁移任务的定时时间，则迁移任务会在设定的时间开始排队并执行，如果没有设置定时任务，则迁移任务会立即执行。

迁移启动后，您可以在迁移任务下看到对应的迁移进度信息。



注：由于系统设计限制，一次性提交或排队多个迁移任务将按排队时间串行执行。

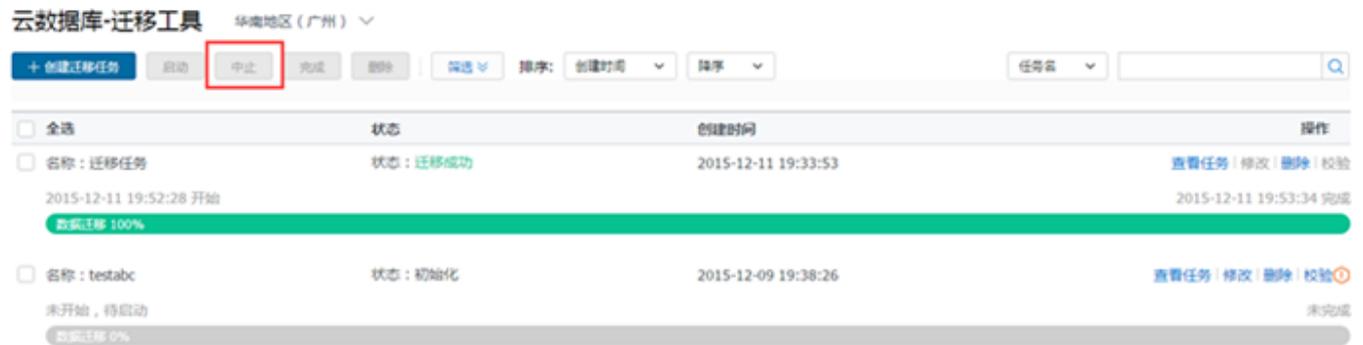
1.5. 增量同步

如果您在创建迁移任务时选择了增量同步选项，那么数据迁移完成后，我们会把目标TDSQL库设置成源数据库的备库，通过主备同步来把迁移过程中源库的新增的数据同步到目标TDSQL库中。此时，源库上的修改都会同步到目标TDSQL中，你可以把业务切换到目标TDSQL上，然后点击完成迁移。

点击完成迁移后，主备同步关系会断开。

1.6. 中止迁移

在迁移过程中，如果您需要停止迁移，可以点击中止按钮进行中止。



注意：再次启动可能导致校验失败或任务失败，您可能需要手动清空目标库内的数据，才能再次启动迁移任务。

2.通过第三方工具迁入

如果您的源库不在腾讯云上或者在VPC网络环境内，那么您可以通过第三方工具来先把源库的数据导出，然后上传到腾讯云服务器上，再把数据导入到TDSQL。

常见的导入导出工具有：mysqldump、mydumper、MySQL GUI Tools等。

数据迁出

您可以通过第三方工具如mysqldump、mydumper来把TDSQL中的数据导出到目标位置。

解压备份和日志文件

出于压缩性能和压缩比的综合考虑，TDSQL的备份文件和日志文件（binlog文件）采用[LZ4 \(Extremely Fast Compression algorithm\)](#)

工具进行压缩，您可以选用LZ4工具进行解压。由于常见的解压工具不支持该格式，本文特别给出解压工具和操作指引。

Windows

下载工具

点击[这里](#)下载解压工具。

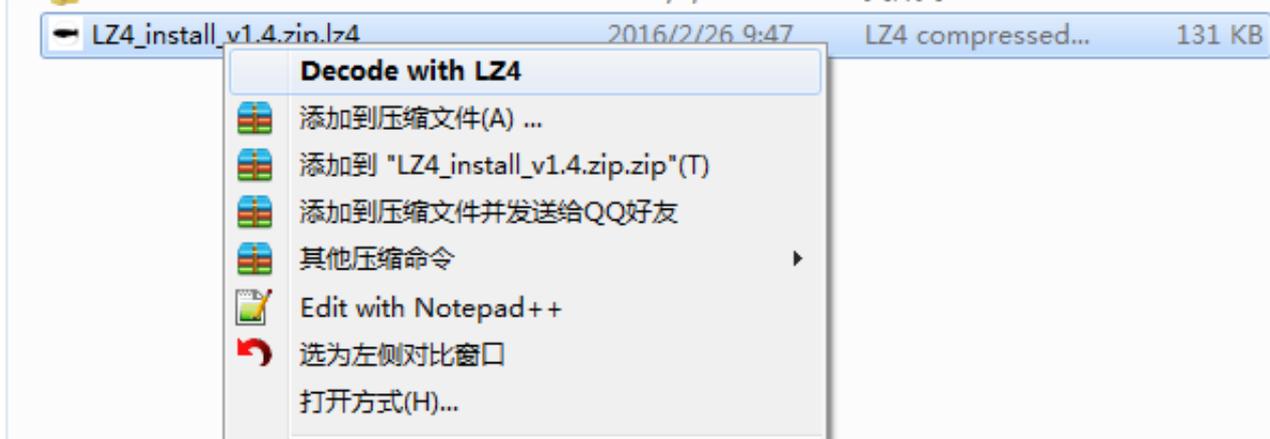
安装工具

双击zip文档，解压后得到LZ4installv1.4.exe，双击运行，然后按指引完成安装。

如果只是用来解压我们的文件，最后一步的复选框可以忽略。

解压文件

如下图，右键点击需要解压的lz4文件，选择Decode with LZ4项即可完成解压



Linux

安装工具

腾讯云的云主机的yum库中有LZ4组件，直接执行如下命令即可安装：

```
$ yum install lz4
```

直接执行lz4返回类似如下图所示，表示安装正确：

```
[root@VM_240_177_centos ~]# lz4
Incorrect parameters
Usage :
    lz4 [arg] [input] [output]

input   : a filename
          with no FILE, or when FILE is - or stdin, read standard input
Arguments :
  -1     : Fast compression (default)
  -9     : High compression
  -d     : decompression (default for .lz4 extension)
  -z     : force compression
  -f     : overwrite output without prompting
  -h/-H : display help/long help and exit
```

解压文件

执行如下命令即可完成解压：

```
$ lz4 -d xxx.lz4
```

CLI参考

TDSQL基于MariaDB10.0.10，兼容MySQL5.5的SQL语法，如有需要请查阅MySQL5.5或者MariaDB10.0.10手册。

恢复备份

TDSQL可以通过回档功能来查看历史数据，如果您需要在自己的机器上进行恢复，可以按照本文档的步骤进行操作恢复历史数据。

1、准备恢复所需环境

注意：请确保使用处于同一个基础网络或私有网络的云主机，以保障网络正常连通

基本配置建议：CPU：2核或以上；内存：4G或以上；磁盘空间：必须超过数据库已用空间；操作系统：centos最新版

基本环境需要：Mariadb 10.0.10

安装过程举例：

- 添加yum源

```
vi /etc/yum.repos.d/mariadb-10.0.10.repo):  
# MariaDB 10.0 CentOS repository list - created 2016-05-30 02:16 UTC  
# http://downloads.mariadb.org/mariadb/repositories/  
[mariadb]  
name = MariaDB  
# baseurl = http://yum.mariadb.org/10.0/centos7-amd64  
baseurl = http://archive.mariadb.org/mariadb-10.0.10/yum/centos6-amd64/  
gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB  
gpgcheck=0
```

- 检查配置yum源对应的mariadb版本是否为10.0.10

```
yum makecache  
yum info MariaDB-server
```

- 安装MariaDB-server

```
yum install MariaDB-server
```

注：如果有提示和以前安装的版本冲突，您可能需要先移除以前安装的包，例如: `yum remove mariadb-libs`

- 安装mariadb客户端

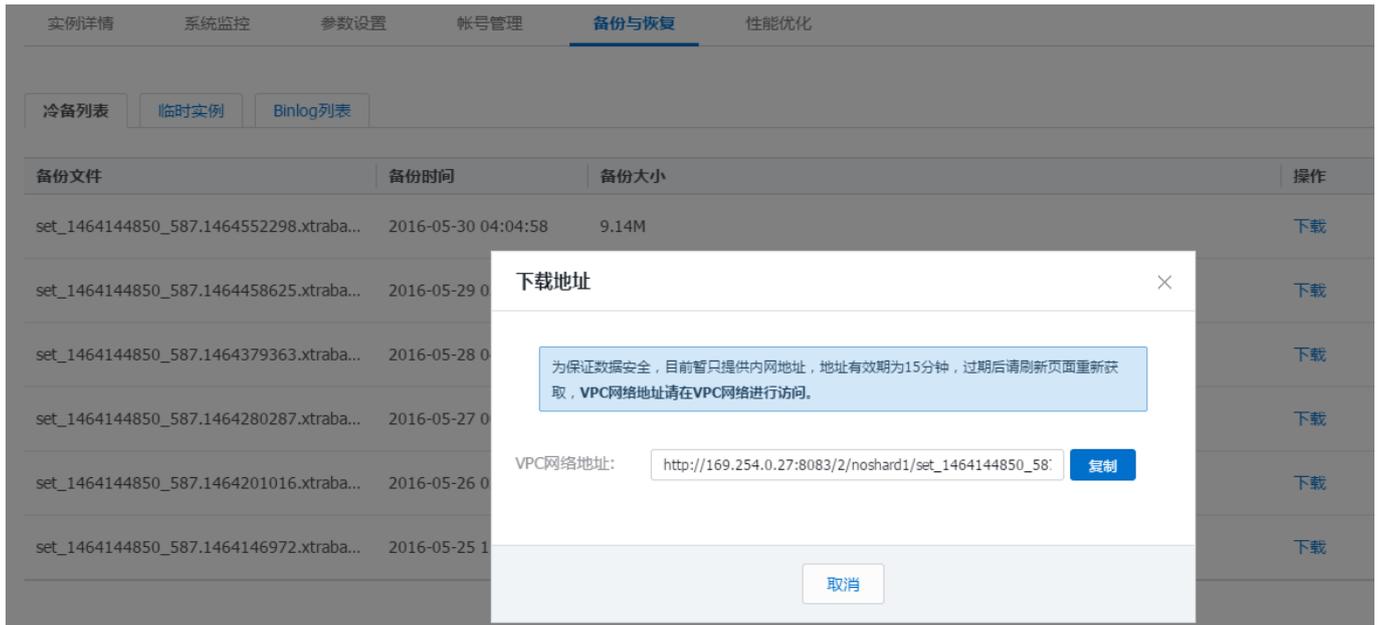
```
yum install MariaDB-client
```

- 安装lz4解压软件

```
yum install -y lz4
percona-xtrabackup
yum install http
://www.percona.com/downloads/percona-release/redhat/0.1-3/percona-
release-0.1-3.noarch.rpm
yum install percona-xtrabackup
```

2、下载备份

首先在“腾讯云管理中心>云数据库>管理中心>备份与恢复”获取备份下载地址



下载命令示例

```
wget --content-disposition
' http://169.254.0.27:8083/2/noshard1/set_1464144850_587/1464552298xxxxxxxx'
```

3、恢复备份文件

- 1) 进入备份文件下载目录，解压通过lz4解压冷备文件

```
lz4 -d set_1464144850_587.1464552298.xtrabackup.lz4
```

- 2) 使用xbstream工具解压到临时目录xtrabackuptmp中

```
mkdir xtrabackuptmp/
mv set_1464144850_587.1464552298.xtrabackup xtrabackuptmp/
xbstream -x < set_1464144850_587.1464552298.xtrabackup
```

解压完后，目录和文件情况大概如下：

```
[root@VM_0_2_centos xtrabackuptmp]# ll
total 6347108
-rw-r----- 1 root root      358 May 30 16:28 backup-my.cnf
-rw-r--r-- 1 root root 1073741824 May 30 16:31 ib_logfile0
-rw-r--r-- 1 root root 1073741824 May 30 16:30 ib_logfile1
-rw-r--r-- 1 root root 1073741824 May 30 16:31 ib_logfile2
-rw-r--r-- 1 root root 1073741824 May 30 16:31 ib_logfile3
-rw-r----- 1 root root 2147483648 May 30 16:31 ibdata1
drwx----- 2 root root      4096 May 30 16:28 mysql
drwx----- 2 root root      4096 May 30 16:28 performance_schema
-rw-r--r-- 1 root root 2149044297 May 30 16:27 set_1464144850_587.1464552298.xtrabackup
drwx----- 2 root root      4096 May 30 16:28 sysdb
drwx----- 2 root root      4096 May 30 16:28 test
-rw-r----- 1 root root        25 May 30 16:28 xtrabackup_binlog_info
-rw-r--r-- 1 root root         58 May 30 16:30 xtrabackup_binlog_pos_innodb
-rw-r----- 1 root root        117 May 30 16:30 xtrabackup_checkpoints
-rw-r----- 1 root root         858 May 30 16:28 xtrabackup_info
-rw-r----- 1 root root 2097152 May 30 16:30 xtrabackup_logfile
[root@VM_0_2_centos xtrabackuptmp]#
```

• 3) 应用日志

使用innobackupex应用日志

```
mkdir /root/dblogs_tmp
innobackupex --apply-log --use-memory=1G --tmpdir=
'/root/dblogs_tmp/' /root/xtrabackuptmp/
```

成功操作后，会显示completed OK!信息，如下所示:

```
InnoDB: Initializing buffer pool, size = 1.0G
InnoDB: Completed initialization of buffer pool
InnoDB: Setting log file ./ib_logfile101 size to 1024 MB
InnoDB: Progress in MB: 100 200 300 400 500 600 700 800 900 1000
InnoDB: Setting log file ./ib_logfile1 size to 1024 MB
InnoDB: Progress in MB: 100 200 300 400 500 600 700 800 900 1000
InnoDB: Setting log file ./ib_logfile2 size to 1024 MB
InnoDB: Progress in MB: 100 200 300 400 500 600 700 800 900 1000
InnoDB: Setting log file ./ib_logfile3 size to 1024 MB
InnoDB: Progress in MB: 100 200 300 400 500 600 700 800 900 1000
InnoDB: Renaming log file ./ib_logfile101 to ./ib_logfile0
InnoDB: New log files created, LSN=210681783
InnoDB: Highest supported file format is Barracuda.
InnoDB: 128 rollback segment(s) are active.
InnoDB: Waiting for purge to start
InnoDB: 5.6.24 started; log sequence number 210681868
xtrabackup: starting shutdown with innodb_fast_shutdown = 1
InnoDB: FTS optimize thread exiting.
InnoDB: Starting shutdown...
InnoDB: Shutdown completed; log sequence number 210683014
160530 16:31:18 completed OK!
[root@VM_0_2_centos ~]# ls
```

- 4) 停止数据库，清空数据文件

停止数据库

```
service mysql stop
```

清空数据文件

```
mkdir /var/lib/mysql-backup  
mv /var/lib/mysql/* /var/lib/mysql-backup
```

- 5) 修改数据库参数文件(/

etc/my.cnf.d/server.cnf)，具体参数数值参考步骤解压开的backup-my.cnf的参数:

注：不能直接用backup-my.cnf替换参数文件。

```
[mysqld]  
skip-name-resolve  
datadir=/var/lib/mysql  
innodb_checksum_algorithm=innodb  
innodb_log_checksum_algorithm=innodb  
innodb_data_file_path=ibdata1:2G:autoextend  
innodb_log_files_in_group=4  
innodb_log_file_size=1073741824  
innodb_page_size=4096  
innodb_log_block_size=512  
innodb_undo_tablespaces=0
```

- 6) 使用innobackupex加载镜像

```
innobackupex --defaults-file=/etc/my.cnf --move-back /root/xtrabackuptmp/
```

成功后，显示Complete OK!, 信息展示如下:

```

160530 15:17:36 [01] Moving ./performance_schema/mutex_instances.frm to /var/lib/mysql/performance_schema/mutex_instances.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_statements_history_long.frm to /var/lib/mysql/performance_schema/events_statements_history_long.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_waits_summary_by_thread_by_event_name.frm to /var/lib/mysql/performance_schema/events_waits_summary_by_thread_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_stages_summary_by_user_by_event_name.frm to /var/lib/mysql/performance_schema/events_stages_summary_by_user_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_statements_summary_global_by_event_name.frm to /var/lib/mysql/performance_schema/events_statements_summary_global_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/users.frm to /var/lib/mysql/performance_schema/users.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_statements_history.frm to /var/lib/mysql/performance_schema/events_statements_history.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_waits_summary_by_host_by_event_name.frm to /var/lib/mysql/performance_schema/events_waits_summary_by_host_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/socket_summary_by_instance.frm to /var/lib/mysql/performance_schema/socket_summary_by_instance.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_statements_summary_by_digest.frm to /var/lib/mysql/performance_schema/events_statements_summary_by_digest.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_waits_summary_by_user_by_event_name.frm to /var/lib/mysql/performance_schema/events_waits_summary_by_user_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_waits_summary_by_account_by_event_name.frm to /var/lib/mysql/performance_schema/events_waits_summary_by_account_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/events_stages_summary_by_host_by_event_name.frm to /var/lib/mysql/performance_schema/events_stages_summary_by_host_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 [01] Moving ./performance_schema/file_summary_by_event_name.frm to /var/lib/mysql/performance_schema/file_summary_by_event_name.frm
160530 15:17:36 [01] ..done
160530 15:17:36 completed OK!
    
```

- 7) 启动数据库

```

chmod 777 -R /var/lib/mysql
service start mysql
    
```

注：如果遇到启动失败，则需要检查错误信息，修复后再重新启动

- 8) 连接数据库查看数据

数据库启动后，您可能需要通过原来的账号密码来连接数据库查看数据。

数据库审计

1 概述

1.1 背景说明

企业使用数据库，可能面临如下安全风险，该类风险需要完整的事后审计和追溯机制，数据库审计能力就由此诞生。

管理风险

- 系统管理员存在的误操作、违规操作、越权操作，损害业务系统安全运行；
- 多人公用一个帐号，责任难以分清；
- 第三方开发维护人员的误操作，恶意操作和篡改；
- 超级管理员权限过大，无法审计监控；

技术风险

- 应用系统开发商后门或漏洞；
- 离职员工留下后门。

政策风险

- 无法达到国家等级保护（三级）明确要求（7.1.3.3）；
- 满足不了行业信息安全合规性文件要求——如人行《金融行业信息系统信息安全等级保护实施指引》；

1.2 术语定义

审计策略：定义对哪些用户行为进行审计以及如何响应的策略。

【审核策略】=【审核对象】+【审计规则】+【响应动作】即配置一条审计策略，需要指定审计内容，如果经过解析，某些（用户或系统）行为的特征正好符合某个审计规则，且恰好在策略生效时间，审计引擎就会按照此策略定义的响应方式进行响应，例如告警等。

审计规则：审计策略中，规定了需要审计的一系列行为的集合，称为规则。规则由规则参数组成，每个规则参数定义了一种具体的行为匹配特征。

1.3 产品能力于限制条件

腾讯云提供数据库审计能力，审计日志默认保存15天（后续版本可延长保存时间），帮助企业对可能存在的数据库访问进行风险控制，提高数据安全等级。

2 审计操作

2.1 开通数据库审计

目前，凡使用云数据库（TDSQL）的用户，均可免费开通数据库审计；开通入口在“腾讯云管理中心>云数据库>TDSQL>数据库审计”页面。

开通审计存在以下注意事项，请知悉：

- 您必须至少拥有1个CDB for TDSQL实例，且未下线或隔离；否则系统会自动关闭您的审计功能。
- 2016年6月5日之前购买的TDSQL实例，需重启升级后方可支持该能力，由于重启升级可能会导致1~5秒的业务中断，您可以联系腾讯云工作人员预约升级时间。
- 数据库审计日志将以明文形式展示，因此建议您开启[二次认证](#)，当然，这不是必选项。



开通审计界面



开通审计可能存在几分钟初始化时间，请耐心等待。

2.2 新建审计规则

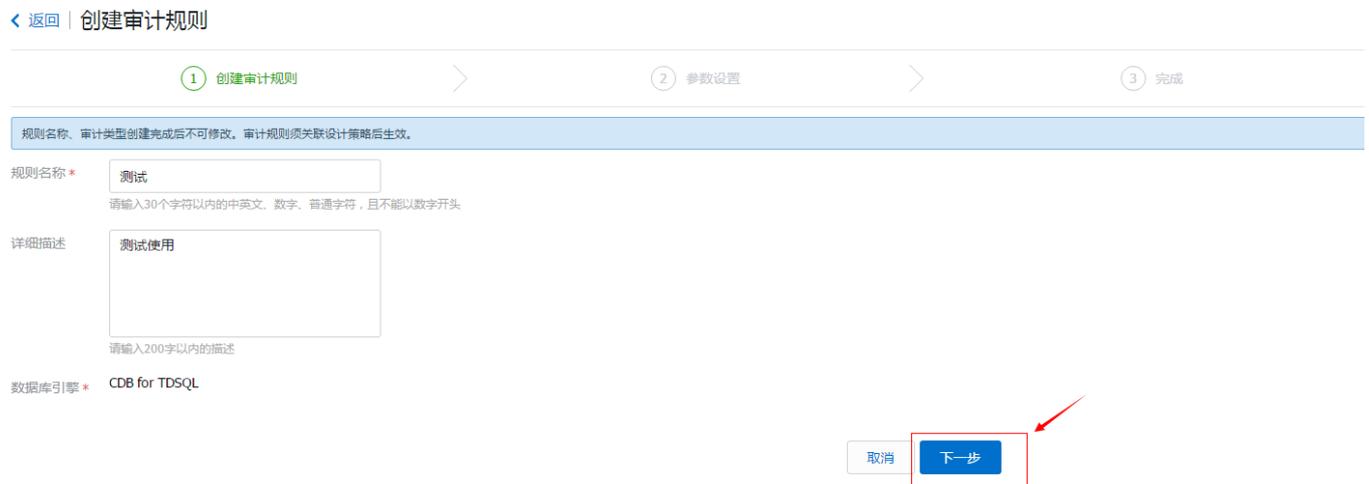
审计功能开通后，日志会自动通过TDSQL网关集群转发到审计集群，但由于没有建立审计规则和审计策略，日

志不会持久化记录并展示。因此您可以通过【新建审计规则】>【关联审计策略】让日志存储在审计集群中。

1、进入审计界面，选择"审计规则"按钮，新建规则



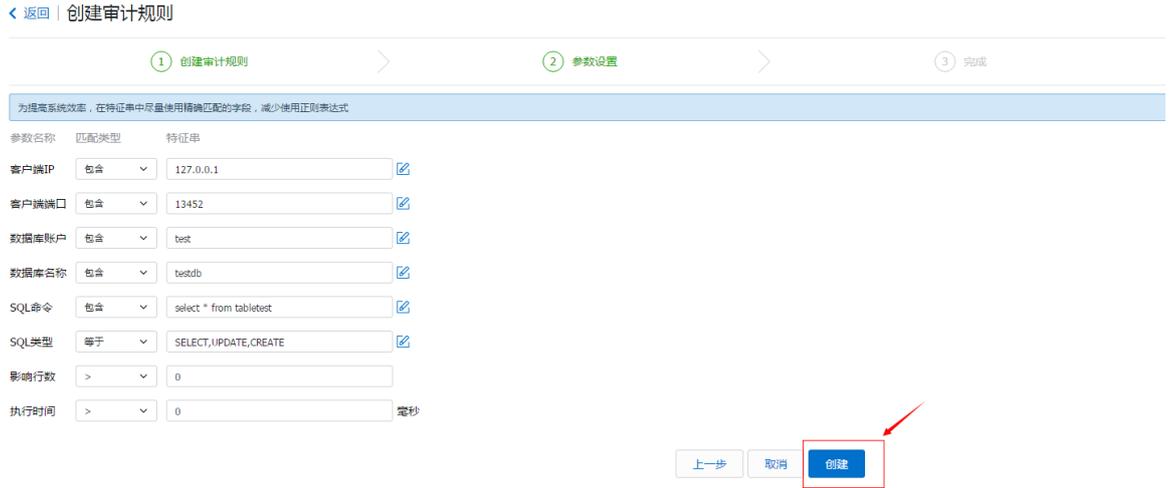
2、填写审计规则名称，选择下一步



3、进入参数设置页面，填写规则参数（所列规则参数需至少填写一个，但不必全部填写）。

填写规则参数需注意以下能力：

- . 规则中参数的条件关系：
 - 与关系；规则中各参数之间是与的关系，即各参数都满足条件规则才会匹配成功。
- . 特征串：定义参数的具体内容，也就是操作对象的具体特征。为了达到精确匹配，用户只定义自己关心的参数的关键字，这样审计系统就只需记录用户定义的规则，提高审计检索效率。注意：特征串为空表示不关心该参数，即"匹配所有"。
- . 匹配类型：参数对象和特征的关系。。
 - . 包含：表示网络字段中出现了特征串就匹配成功；
 - . 不包含：表示网络字段中没有出现特征串则匹配成功；
 - . 等于：表示网络字段等于特征串则匹配成功；
 - . 不等于：表示网络字段不等于特征串则匹配成功；
 - . 正则表达式：则表示特征串，支持标准的[正则表达式语法](#)。



4、所有新建规则，可以在规则列表中看到



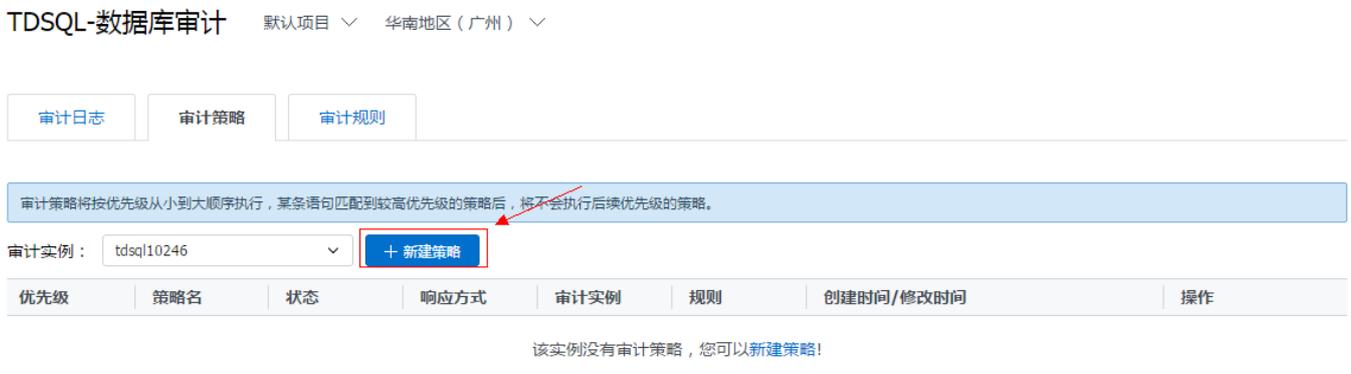
5、设置完成审计规则后，您可以随时“修改”。类似的规则，您可以通过“克隆规则”的方式新建多条审计策略，以提高您的效率。

2.3 新建审计策略

审计策略是将“审计规则、审计对象、响应方式”组合起来形成完整的审计方案。用户可以对一个实例同时制定多条审计策略，审计引擎解析时，将按用户的策略配置顺序从前到后的优先级匹配。

创建审计策略的方式如下：

1、进入策略生成窗：选择“审计策略”点击“新建策略按钮”



2、填写策略要求：根据需求选择需审计实例，并选择对应规则（目前暂不支持配置告警）。

新建策略
×

新建或修改审计策略，预计5分钟后生效，原有审计策略的审计日志不会被清除或修改。

审计实例 tdsql10246

策略名称 测试策略
仅支持长度小于30的中文/英文/数字/"-"/"_"

选择规则 测试
请选择规则

响应动作 放行且记录

警告策略 无

状态

确定
取消

3、调整优先级：对于相同实例下的多条策略，可以进行优先级调整；优先级数字越小，优先等级越高。优先级调整后，预计1分钟内生效。

审计策略将按优先级从小到大顺序执行，某条语句匹配到较高优先级的策略后，将不会执行后续优先级的策略。

审计实例：tdsql10246 + 新建策略

优先级	策略名	状态	响应方式	审计实例	规则	创建时间/修改时间	操作
1	测试策略	启动	记录/放行	tdsql10246	测试	2016-06-12 14:30:35 / 2016-06-12 14:30:...	^ v 修改 删除
2	测试二	启动	记录/放行	tdsql10246	测试	2016-06-12 14:38:04 / 2016-06-12 14:38:...	^ v 修改 删除

4、您可以通过修改功能，实时修改审计策略；修改完成后预计5分钟内生效，并按新策略进行批评，修改审计策略前的日志不会被修改。

2.4 查看日志

匹配到审计策略的SQL语句将展现在审计日志页面，您可以直接点击查看或搜索。注意事项：

- 因设计要求，审计日志明文方式展现，我们再次建议您开启[二次认证](#)，以确保日志可控。
- 日志将以审计策略创建时开始记录，历史数据不做记录。
- 事务、存储过程等可能会被记录为单条语句，详见[数据库审计已支持语法说明](#)

- 目前支持单条SQL语句最大1k，超过部分会被截断。

数据库审计已支持语法

通过测试，数据库审计目前已经支持绝大多数SQL语句，如有发现存在不足，请联系腾讯云工作人员或发送邮件到vitosu@tencent.com反馈。

1、已支持DCL,DDL,DML语句的解析

```
Insert, Replace, Select, Union, Update, Delete
, CreateDatabase:, CreateEvent, CreateFunction, CreateIndex, CreateLog,
CreateTable, CreateServer, CreateProcedure, CreateTablespace, CreateTrigger, Cre
ateView, CreateUDF, CreateUser,
ShowCharset, ShowCollation, ShowColumns, ShowCreate, ShowCreateDatabase, ShowDat
abases, ShowEngines, ShowErrors,
ShowEvents, ShowFunction, ShowGrants, ShowLogEvents, ShowLogs, ShowProcedure, Sho
wOpenTables, ShowPlugins,
ShowProcessList, ShowMasterStatus, ShowPrivileges, ShowProfiles, ShowSlaveHosts
, ShowSlaveStatus, ShowTableStatus,
ShowWarnings, ShowVariabl
es, ShowStatus, ShowTriggers, Call
, DropProcedure, DropDatabase, DropEvent, DropFunction,
DropIndex, DropLogfile, DropServer, DropTables, DropTablespace, DropTrigger, Drop
User, DropView, AlterDatabase,
AlterEvent, AlterFunction, AlterLogfile, AlterProcedure, AlterServer, AlterTable
, AlterTablespace, AlterUser,
AlterView, Rollback, Commit, Begin, Set
, SetTrans, SetPassword, Release, Grant, RenameTable, RenameUser, Revoke,
Install, StopSlave, StartSlave, StartTrans, Use
, DescribeTable, DescribeStmt, Flush, Load, LoadIndex, FlushTables,
Reset, CacheIndex, TruncateTable, Lock, Unlock, SavePoint, Help, Do
, SubQuery, ShowTables, Execute, Deallocate, Binlog,
Kill, Partition
, PrepareRepairXACheckCheckSumAnalyzeChangeOptimizePurgeHandlerSignalResigna
l
```

2、事务和存储过程可能被拆分为多条语句

实例类型

目前TDSQL存在以下实例类型：

- 主实例：（金融级）云数据库CDB for TDSQL直接申请购买的实例，拥有TDSQL的全部功能。
- 灾备实例
：部署在异地（不同机房）与主实例实时同步数据的实例，通常用于数据库容灾，仅监控、备份等功能可用。
- 临时实例
：由回档功能生成的临时存在的实例，用于回档后数据校验，可通过切换为主实例功能，将校验后的临时实例恢复为主实例。
- 逻辑实例：分布式云数据库DCDB for TDSQL在逻辑上的完整的实例，是由多个物理分片组成。
- 物理分片
：通过水平拆分，将逻辑实例均匀的拆分到不同的数据库实例，物理上独立的数据库简称物理分片。