TencentDB for MySQL

Tencent Cloud

(Tencent Cloud) (Tencent) (,
) . 3 .

(Tencent Cloud)
, . . .
,
. .

:

TencentDB for MySQL

MySQL
MyISAM InnoDB
TencentDB for MySQL VPC
TencentDB for MySQL
2 3
/ TencentDB for MySQL
DTS InnoDB RocksDB
LAMP
Drupal
Python MySQL API

TencentDB for MySQL

:: 2024-07-25 16:38:48

- TencentDB for MySQL TencentDB for MySQL
.
- TencentDB for MySQL SQL
.

- TencentDB for MySQL TencentDB for MySQL super, shutdown, file
. TencentDB for MySQL set .

```
#1227-Access denied;you need(at least one of)the SUPER privilege (s)
for this operation
```

: set , [ ]>[
]
.
- . DML(SELECT, UPDATE, INSERT, DELETE)
.
- ,
.
- IP IP ,
. .
IP .
- .

- .
- CVM MySQL
VPC .
- binlog , MySQL TencentDB
for MySQL .

. 3.4 mysqlbinlog .

- CVM / url . 404 .

- online ddl . `pt-online-schema-change` .
- .
.
- .
.
- `begin` . TencentDB for MySQL (5). , . DB ' DB _bak' .
- .
. .

DB

- TencentDB for MySQL 5.6 MyISAM Memory . Memory TencentDB for Redis Memcached . TencentDB for MySQL MyISAM InnoDB .
- , 1 .
- `row_format` .
- .
. MySQL 1 InnoDB .
. `binlog_format` row / .
- NOT NULL . NULL SQL . NULL IS NULL IS NOT NULL .

- (/ QPS, TPS,)
  . TencentDB for MySQL .
- .
  , .
- utf8mb4 .
  utf8mb4 .
  , TencentDB for MySQL utf8mb4 .
- decimal . float double .
  decimal .
- , , , ,
  text/blob . .
- . .
  .
  .
- , .
  , , , , event,
  . ,
  definer . definer
  .
- . , .
  80% .
- , ( )
  / .

- . B+
  . .
- . , `select xxx where a = x and b = x;` a b a `idx_ab(a, b)'를 생성합니다. 비등호와 등호가 혼합된 판단 조건이 있는 경우 반드시 등호 조건 열을 앞에 놓아야 합니다. 예를 들어,` where a xxx and b = xxx` a b . a
  .

- 5 , 5 . , .
- SQL , cardinality . SQL . , cardinality . cardinality 10% (:).
- varchar , . varchar , . . count(distinct left(,))/count(*) .
- . (a,b) (a) , (a) redundant index . a , (a,b) (a) .
- IO . InnoDB 2 . SQL , , . (flashback table) . IO , . , `select a,b from xxx where a = xxx` a a, b . .

SQL

- UPDATE, DELETE LIMIT , WHERE . LIMIT .
- `INSERT INTO t_xxx VALUES(xxx)` . , .
- SQL .
- . a varchar , SQL where a = 1; varchar int .
- . , .
- join .
- . (a,b) SQL order by a b desclike .

- `xx%` ,
  .
- (not, !=, not in)
  .

- `select *`
  .
  - I/O
  - `innodb_buffer_pool_size`
  -
- . /
  .
- .
- join , join . join
  , join .
- LIMIT . LIMIT 80000, 10 80010 10
  , .
  . : `SELECT * FROM test WHERE id >= ( SELECT sql_no_cache id FROM test order by id LIMIT 80000,1 ) LIMIT 10 ;`
- SQL . MySQL 5.5 in exists
  . .

:

- .
  . .
- , `log_queries_not_using_indexes` ,
  . , .
- SQL ,
  .

:: 2024-07-25 16:38:48

.

TencentDB for MySQL

, hang , ,

. .

, /

.

, .

/ TencentDB for MySQL

, connectTimeOut socketTimeOut

.

. OLTP(On-Line Transaction Processing)

20 .

:

- connectTimeOut: TCP

.

- socketTimeOut: TCP SQL

.

MySQL

:: 2024-07-25 16:38:48

TencentDB for MySQL .

. .

lower_case_table_names

: 0
** : ** , .
0, 1 0 . 0
, 1 .
** : ** , .
, (: 2 ,
Test, TEst),
. TEst .

auto_increment_increment

: 1
: AUTO_INCREMENT . 1~65535 ,
1 .
: .
binlog_format statement , .
, /
.

auto_increment_offset

: 1
: AUTO_INCREMENT ( ) . 1~65535 ,
1 .
: .
/ Auto Increment , /
.

sql_mode

: NO_ENGINE_SUBSTITUTION

: MySQL sql mode , sql mysql sql , . 5.6 NO_ENGINE_SUBSTITUTION , Uncompiled . 5.7 `ONLY_FULL_GROUP_BY` , `STRICT_TRANS_TABLES` , `NO_ZERO_IN_DATE` , `NO_ZERO_DATE` , `ERROR_FOR_DIVISION_BY_ZERO` , `NO_AUTO_CREATE_USER` , `NO_ENGINE_SUBSTITUTION` . ,

- `ONLY_FULL_GROUP_BY` GROUP BY SELECT , HAVING ORDER BY , GROUP BY GROUP BY .
- `STRICT_TRANS_TABLES` Strict mode .
- `NO_ZERO_IN_DATE` , 0 , Strict mode .
- `NO_ZERO_DATE` 0 , Strict mode .
- `ERROR_FOR_DIVISION_BY_ZERO` Strict mode INSERT UPDATE 0 , Strict mode 0 MySQL NULL .
- `NO_AUTO_CREATE_USER` GRANT .
- `NO_ENGINE_SUBSTITUTION` Uncompiled .

: .

sql mode , .

sql mode sql mode , SQL / .

MyISAM InnoDB

:: 2024-07-25 16:38:48

MyISAM InnoDB

.

TencentDB for MySQL InnoDB , MySQL 5.6
MyISAM Memory .
.

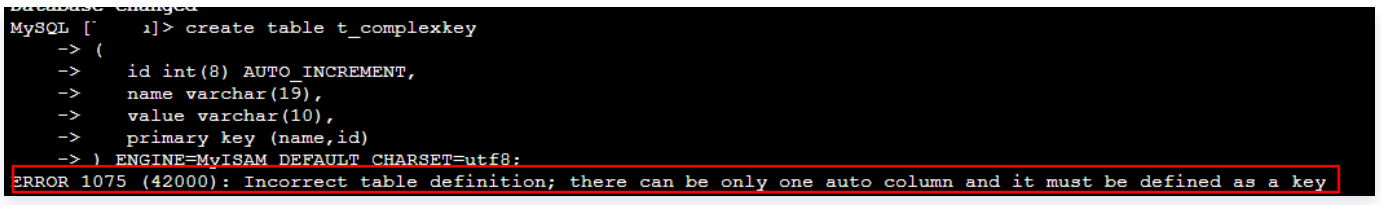
TencentDB for MySQL 5.6 ,
MyISAM InnoDB .
MyISAM Auto Increment Compound Primary Key InnoDB
. MyISAM InnoDB .
. `ERROR 1075 (42000):Incorrect table definition;there can be only one auto column and it must be defined as a key`
Auto Increment InnoDB Auto Increment Compound Primary Key .

InnoDB Auto Increment Compound Primary Key

1. SQL :

```
create table t_complexkey
(
id int(8) AUTO_INCREMENT,
name varchar(19),
value varchar(10),
primary key (name,id),
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

:



2. SQL :

```
create table t_complexkey
(
id int(8) AUTO_INCREMENT,
name varchar(19),
value varchar(10),
primary key (name,id),
key key_id (id)  ## [illegible]
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

:

```
MySQL [    1]> create table t_complexkey
    -> (
    ->    id int(8) AUTO_INCREMENT,
    ->    name varchar(19),
    ->    value varchar(10),
    ->    primary key (name,id),
    ->    key key_id (id)
    -> ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
Query OK, 0 rows affected, 1 warning (0.01 sec)
```

3. :

```
show create table t_complexkey;
```

```
ySQL [huawu]> show create table t_complexkey;
---------------+-------------------------------------------------------------------
-------------------------------+
 Table         | Create Table
                                |
---------------+-------------------------------------------------------------------
-------------------------------+
 t_complexkey | CREATE TABLE `t_complexkey` (
 `id` int(8) NOT NULL AUTO_INCREMENT,
 `name` varchar(19) NOT NULL,
 `value` varchar(10) DEFAULT NULL,
 PRIMARY KEY (`name`,`id`),
 KEY `key_id` (`id`)
 ENGINE=InnoDB DEFAULT CHARSET=utf8 |
---------------+-------------------------------------------------------------------
-------------------------------+
 row in set (0.00 sec)
```

TencentDB for MySQL VPC

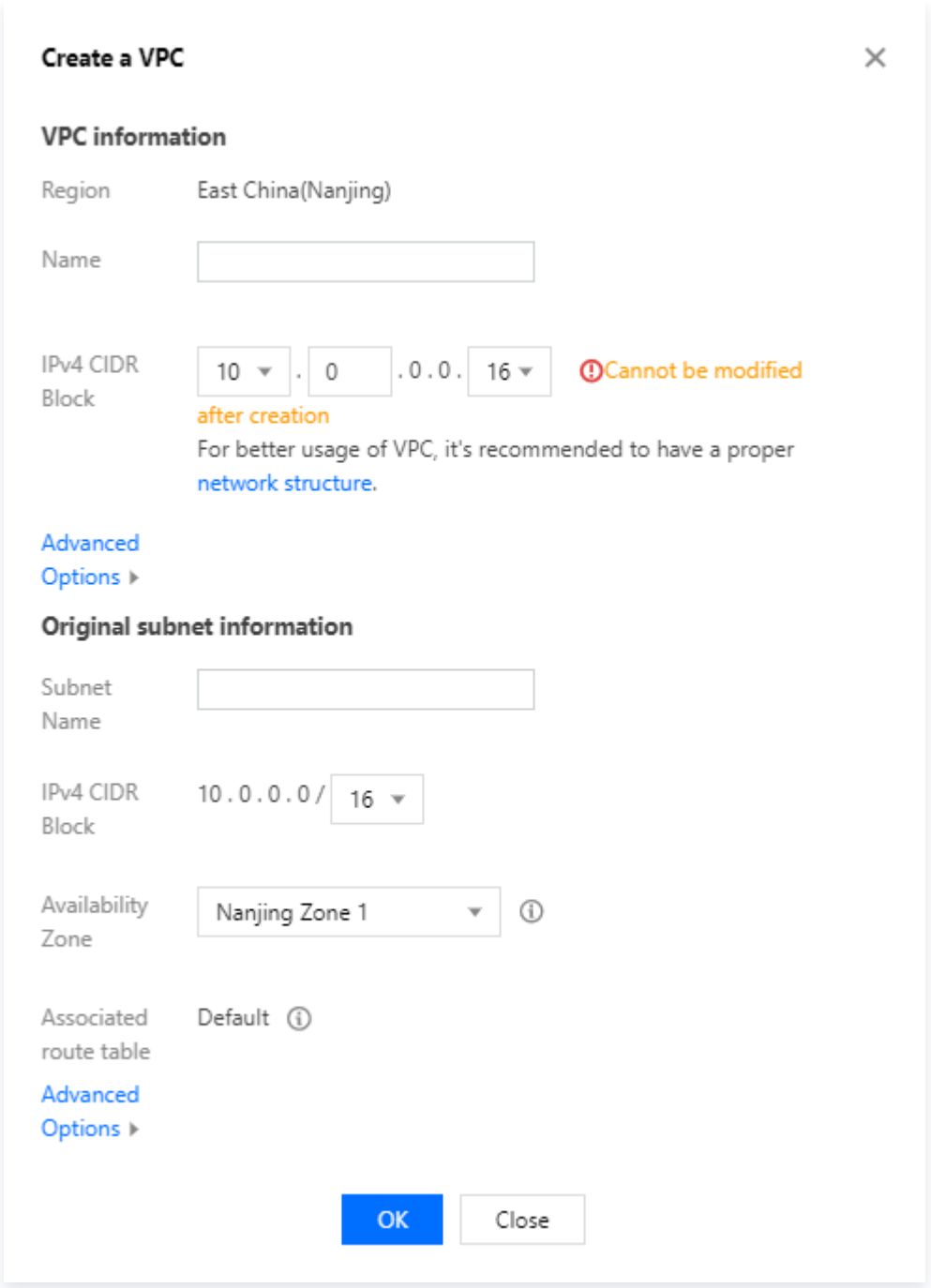
:: 2024-07-25 16:38:48

Tencent Cloud (CDB) : Tencent Cloud VPC
. VPC Tencent Cloud (: Tencent Cloud CDB).
VPC CDB .
VPC , CDB VPC .
VPC CVM TencentDB for MySQL , VPC
.

1 : VPC

VPC 1 , .

1. VPC .
2. VPC +New .
3. VPC . VPC CIDR
.
   - VPC CIDR IP 1 . VPC PC/
     CIDR .
     - `10.0.0.0` ~~`10.255.255.255`~~ ~~(1628)~~
     - `172.16.0.0` ~~`172.31.255.255`~~ ~~(1628)~~
     - `192.168.0.0` ~~`192.168.255.255`~~ ~~(1628)~~
   - CIDR VPC CIDR .
     VPC IP `192.168.0.0/16` , VPC IP `192.168.0.0/16` 、 `192.168.0.0/17` .

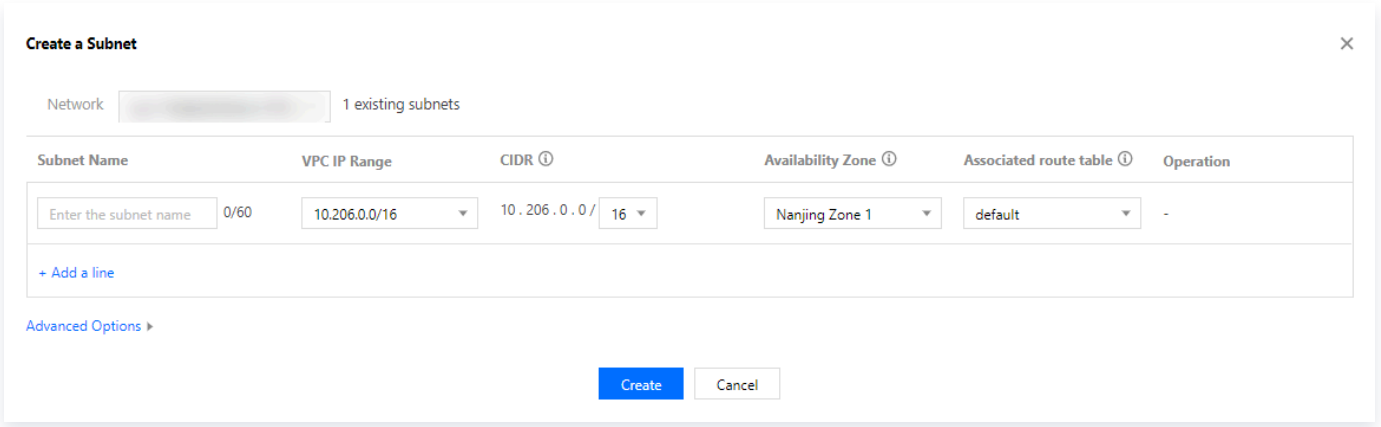


2 :

1 .

1. VPC .
2. .
3. VPC +New .

4. , CIDR, , .



5. ( ) [+] .

6. .

3 :

, , .

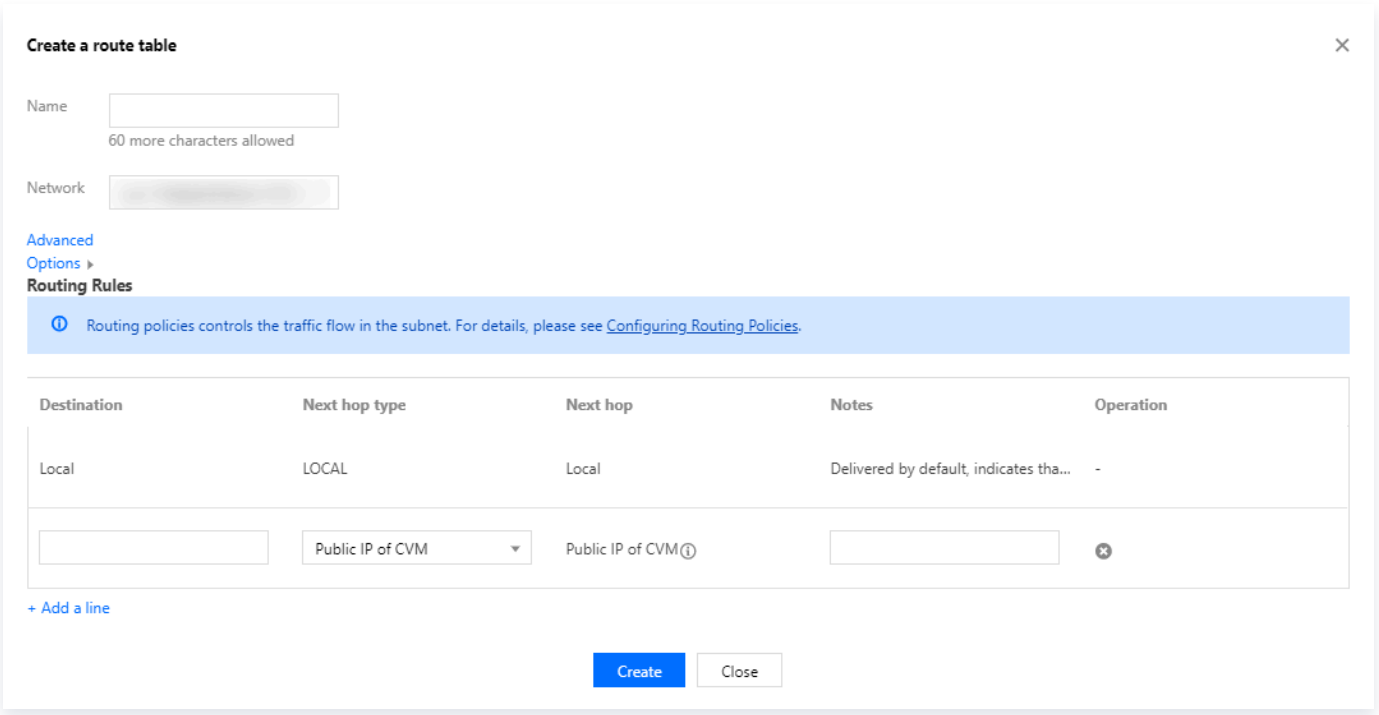
.

1. VPC . .

2. VPC +New .

3. , .

.

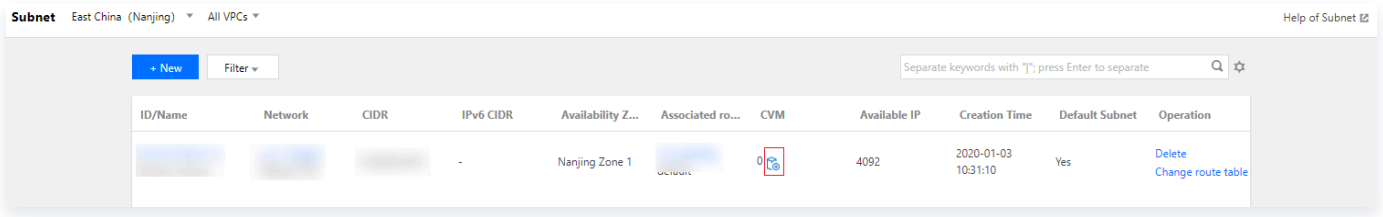


4. [ ] .

.

4 : CVM

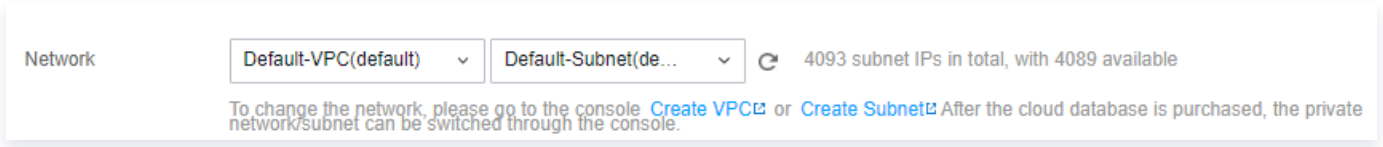
1. VPC .
2. .
3. CVM CVM .



4. CVM . CVM .

5 : CDB

1. TencentDB for MySQL . New .
2. .
, CDB .



1. Operation Manage
.
2. VPC .

TencentDB for MySQL

:: 2024-07-25 16:38:48

.

TencentDB for MySQL QPS

.

1.

1.1

TencentDB for MySQL v5.5, v5.6, v5.7 v8.0 MySQL

. v5.5 ,

v5.6 .

MySQL v5.7 . ,

. .

- JSON
  MySQL v5.7 MySQL JSON

  .
- : JSON JSON . , JSON

  .
- : JSON JSON .

  .
- : query JSON .

  ‘ ’ .
- : JSON features SQL

  . feature JSON .

  ```
  SELECT feature->"$.properties.STREET" AS property_street FROM
  features WHERE id = 121254;
  ```

  MySQL 5.7

  .
- SYS Schema
  MySQL SYS Schema , , ,

  database schema . Performance Schema INFORMATION_SCHEMA

  DBA .

MySQL SYS Schema MySQL 5.7 , FAQ

.

- ?
- CVM ?
- ?
- InnoDB
- InnoDB(Online DDL) : MySQL Buffer Pool size
  . InnoDB UNDO
  (ibdata1)
  . MySQL v5.7 varchar .
  .
- InnoDB : MySQL v5.7 InnoDB .
  90% .
- InnoDB : MySQL InnoDB 25%
  MySQL
  .

MySQL 5.7 MySQL .

1.2

TencentDB for MySQL CPU . CPU
. .
.

Sysbench .
TencentDB for MySQL
.
.
.
.
.
IO .
.

1.3

TencentDB for MySQL , , binlog .

.
.

1.4

TencentDB for MySQL , , ,

.

.

1.5 TencentDB

TencentDB for MySQL . –

binlog . .

.

1.6 TencentDB

TencentDB for MySQL / .

.

1.7 CVM TencentDB for MySQL

CVM TencentDB for MySQL . MySQL

.

2.

/ 4:1 10:1 ,

.

.

TencentDB for MySQL .

.

.

/

.

3. TencentDB

TencentDB for MySQL

.

. Tencent Cloud

MySQL

. .

4. 2 3

TencentDB for MySQL 2 3 .

- TencentDB for MySQL 1 2
  ( ) .
- 2 3 .

5.

M–S( ) .

.

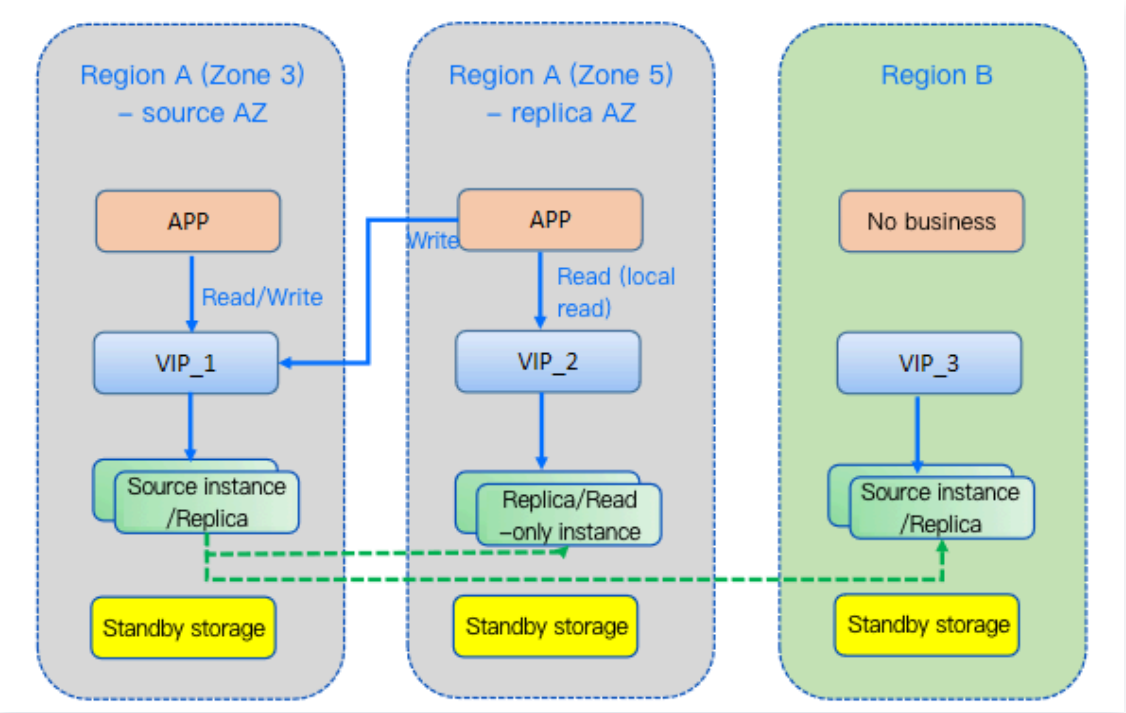
2 3

:: 2025-11-24 20:51:16

(AZ) 2 3

.

2 3

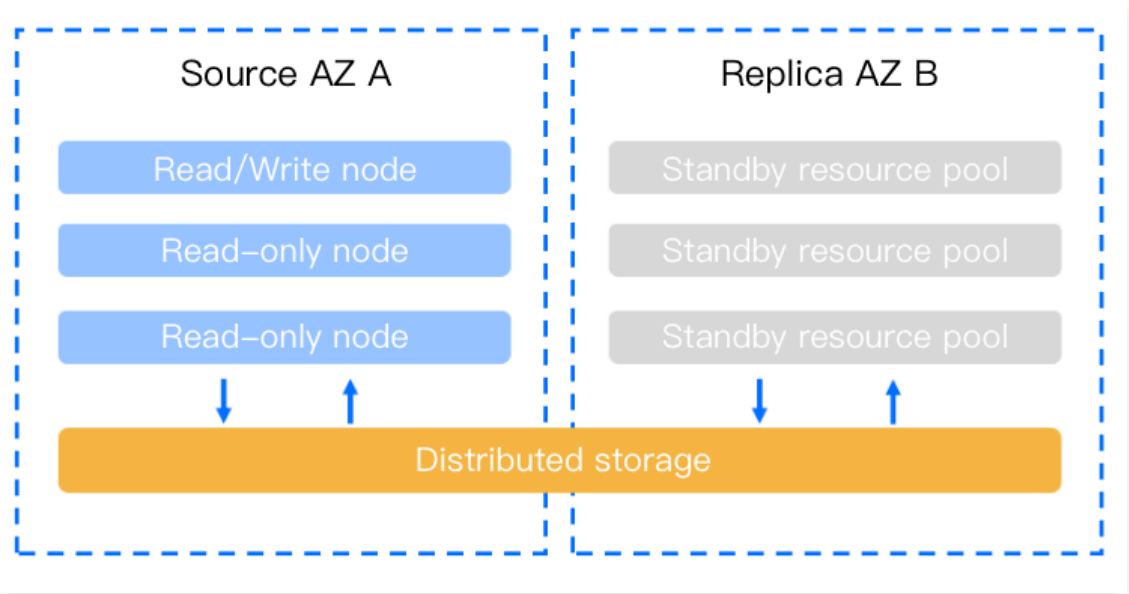


- 1 2 :
  A 3 5 1 2 . 3 5
  .
- :
  A B . A 3, 5
  B .

AZ

TencentDB for MySQL AZ . AZ AZ
, AZ IDC
.

TencentDB for MySQL AZ AZ
.



- .
- 2 AZ .
- AZ .

AZ

TencentDB for MySQL AZ , , , , , , ,
, , , , .
AZ TencentDB for MySQL .

- AZ .
- AZ .

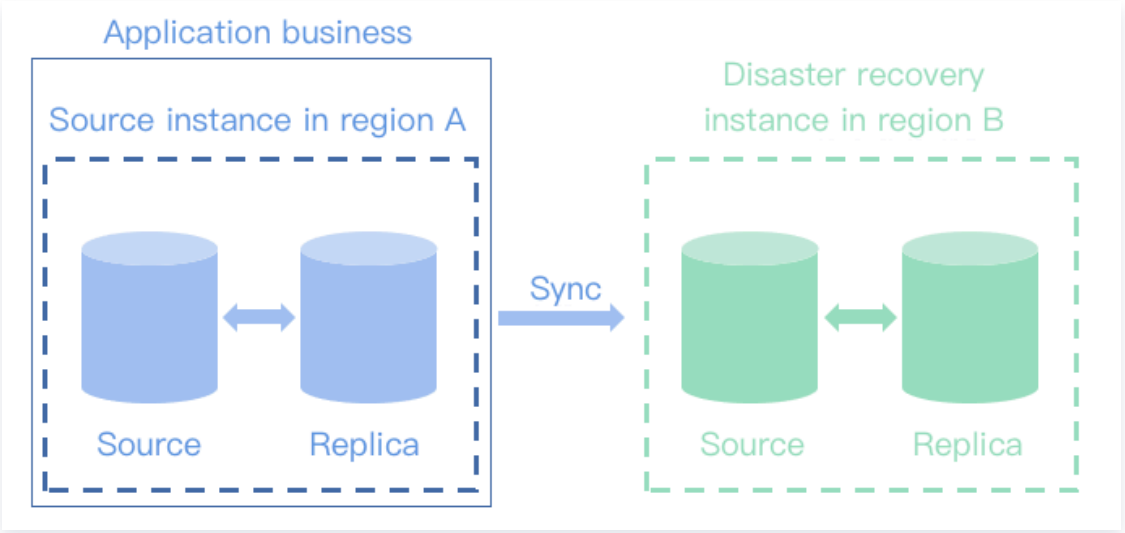
. AZ AZ .

, TencentDB for
MySQL

.

.

/ .



.

2 3

1 : AZ

AZ

1. TencentDB for MySQL
.
2. AZ AZ AZ .

: / AZ 2~3ms .

3. . >
AZ .

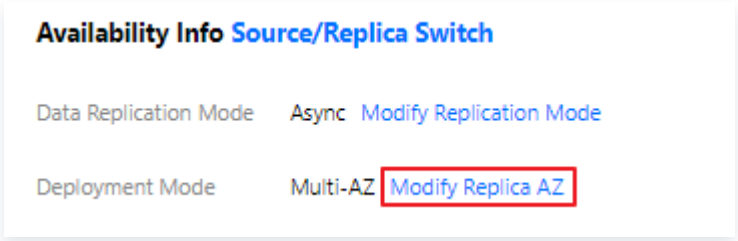
AZ AZ

1. TencentDB for MySQL .

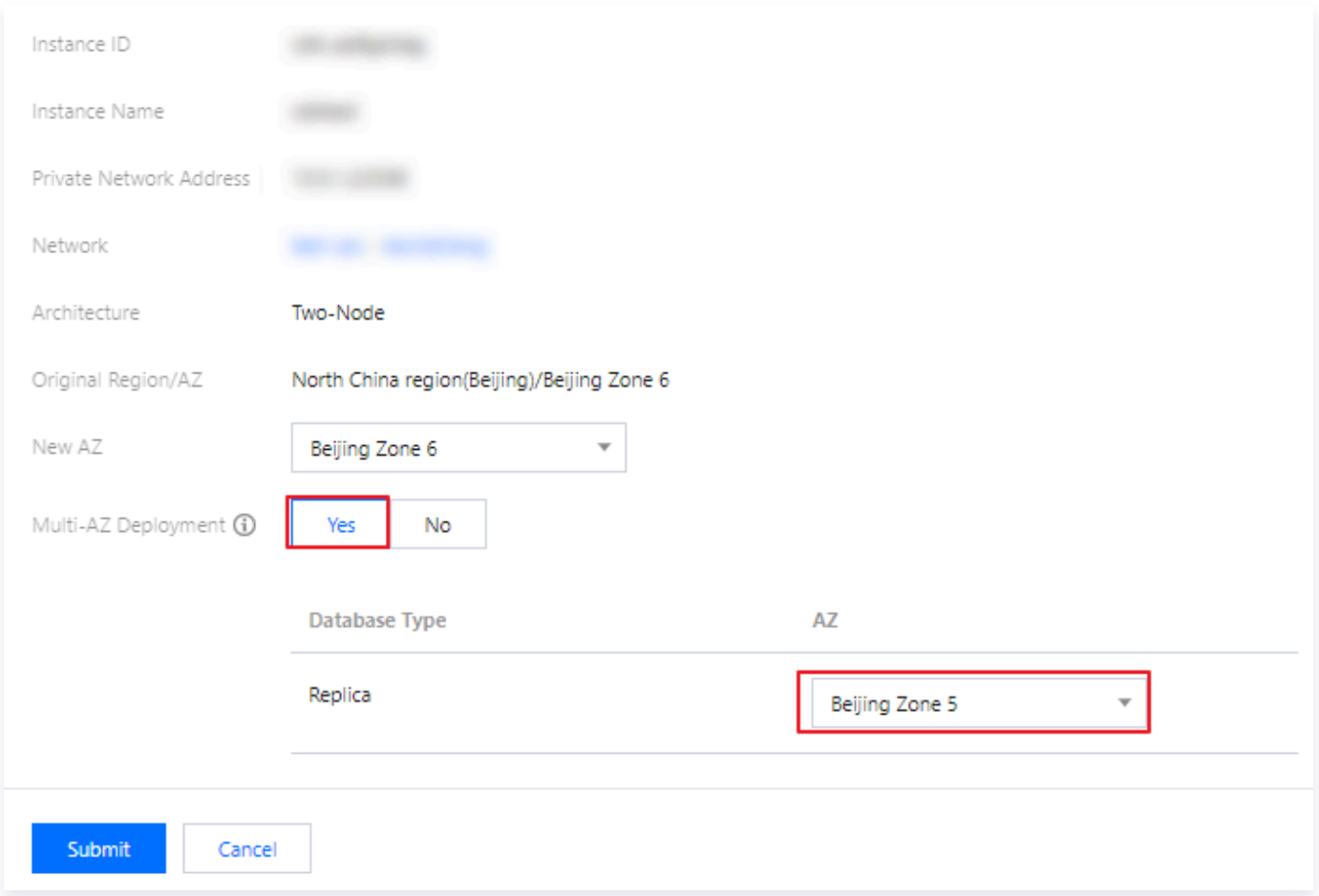
2. ID

.

3. > AZ .



4. AZ AZ .



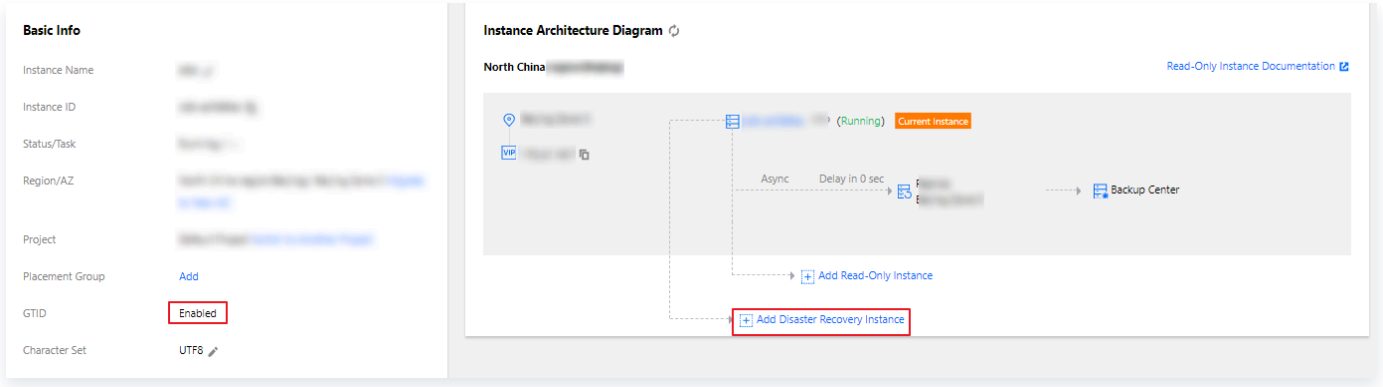
2 :

: 1GB 50GB InnoDB MySQL 5.6 GTID . .

2.1

1. TencentDB for MySQL , ID .

2. GTID .



3. , .

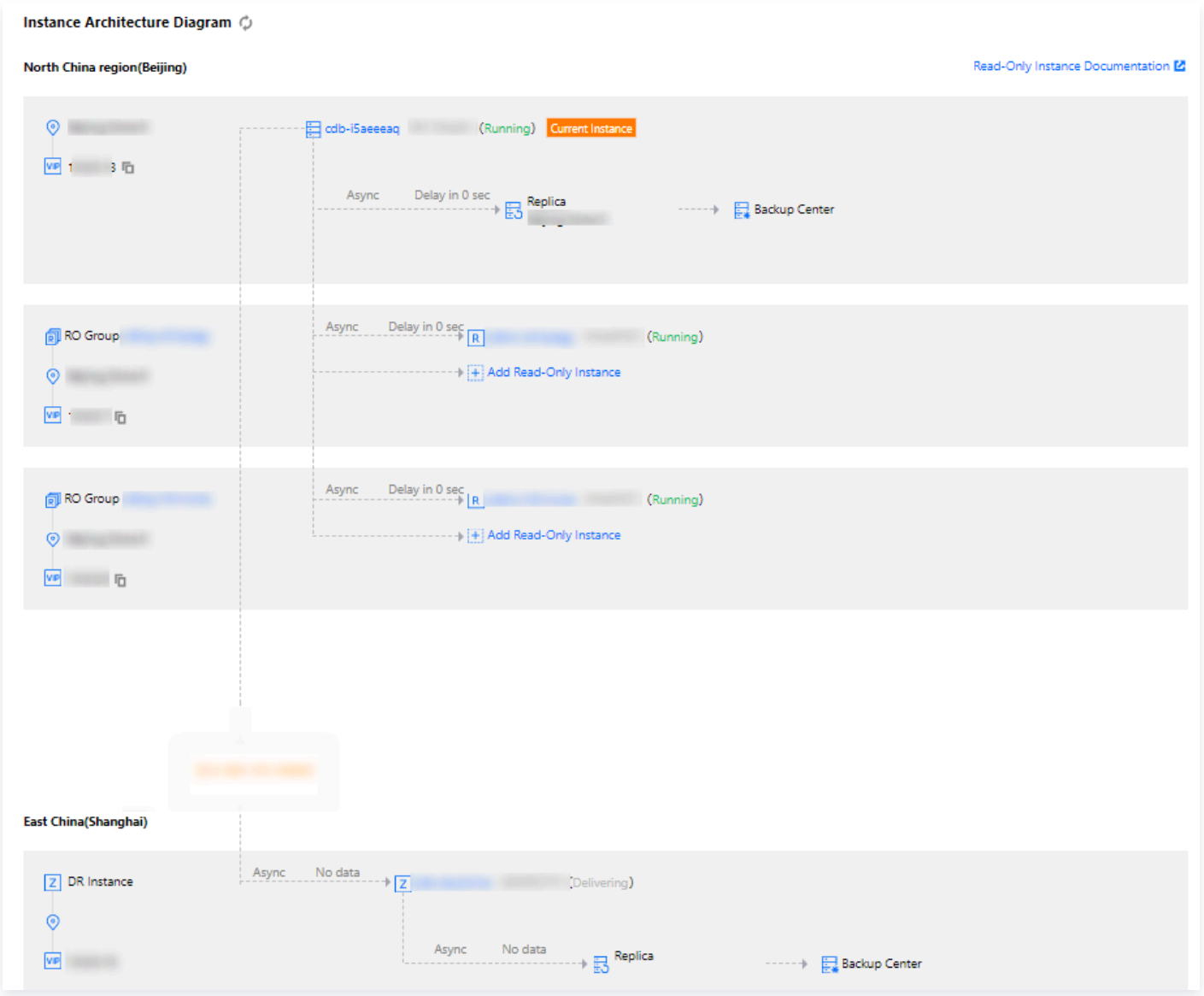
:

4. .

5. . , .

2 3

2 3 .


Instance Architecture Diagram
North China region(Beijing)
Read-Only Instance Documentation
cdb-i5aeeeaq
(Running)
Current Instance
Async
Delay in 0 sec
Replica
Backup Center
RO Group
Async
Delay in 0 sec
(Running)
Add Read-Only Instance
RO Group
Async
Delay in 0 sec
(Running)
Add Read-Only Instance
East China(Shanghai)
DR Instance
Async
No data
(Delivering)
Async
No data
Replica
Backup Center

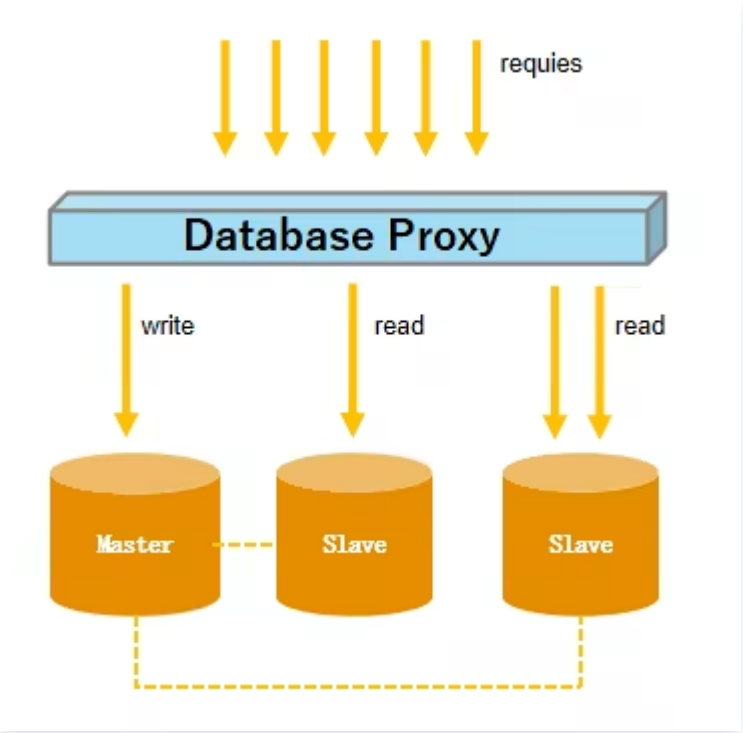
/ TencentDB for MySQL

:: 2024-07-25 16:38:48

/

TencentDB for MySQL .

/



TencentDB .

TencentDB .

. .

. /

.

/

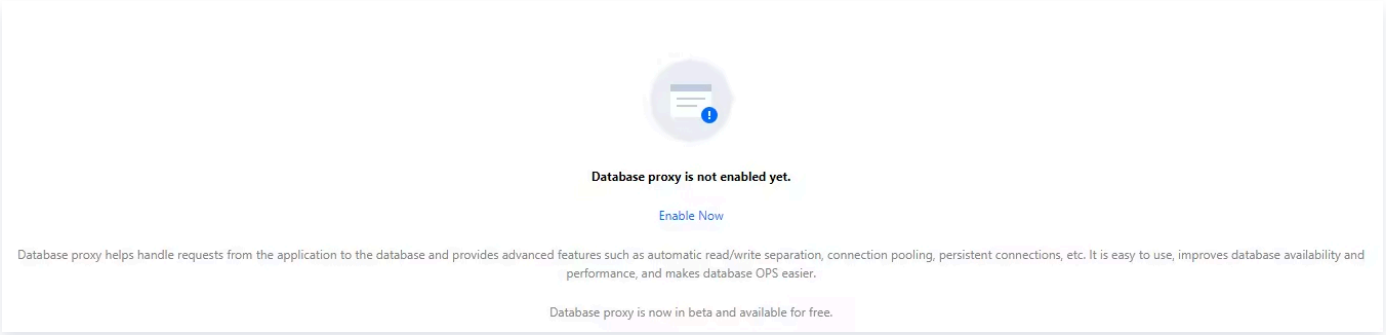
, .

.

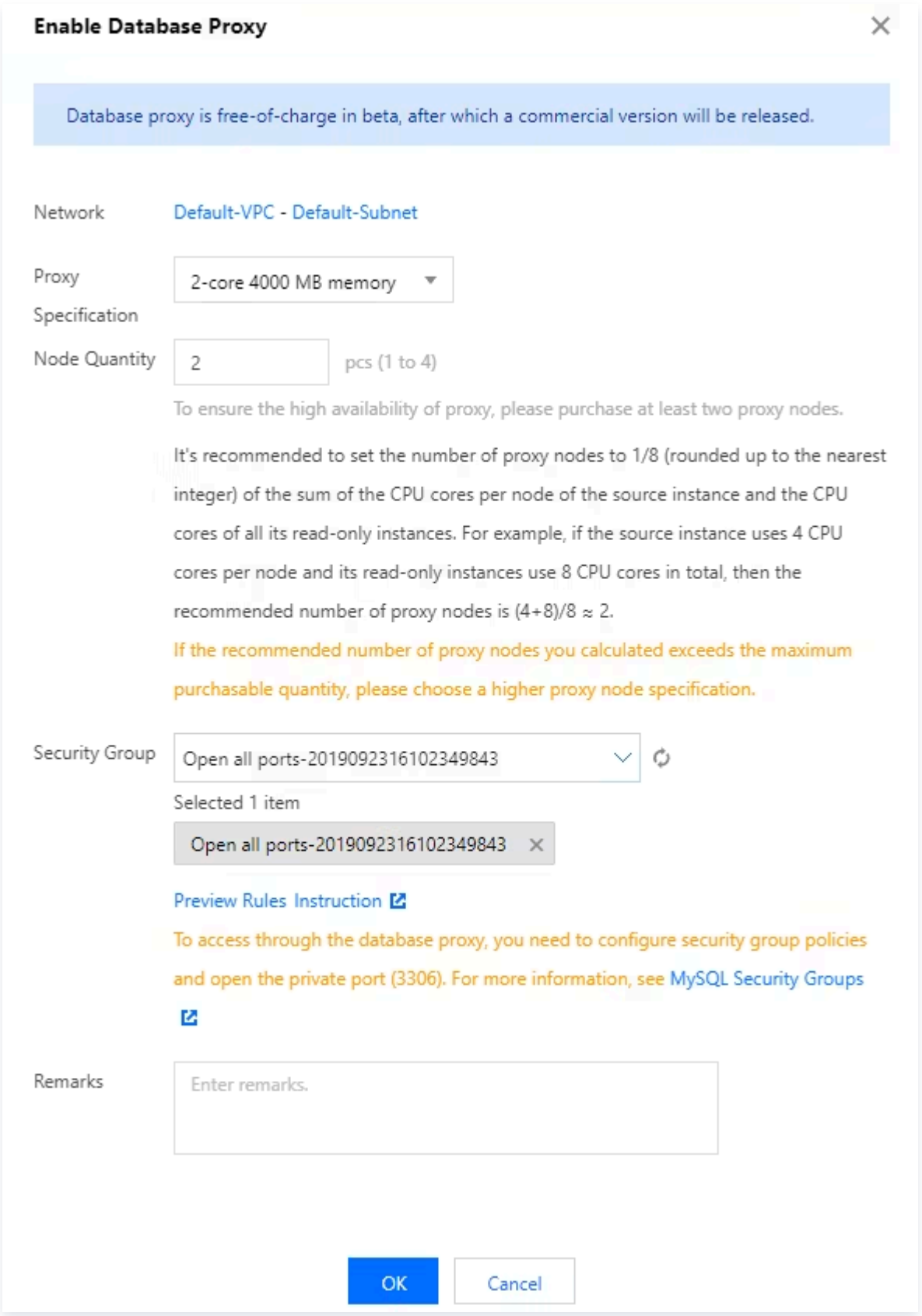
. /
/
.
/
. ,
.
/

1 :

1. TencentDB for MySQL ,
, ID .
2. .



3. .
- : VPC , .
- : 2 4000MB , 4 8000MB , 8 16000MB .
- : . CPU 1/8( ) . : CPU 4 , CPU 8 , = (4 + 8) / 8 ≈ 2
- :
.
- :
.


Enable Database Proxy
Database proxy is free-of-charge in beta, after which a commercial version will be released.
Network
Default-VPC - Default-Subnet
Proxy Specification
2-core 4000 MB memory
Node Quantity
2
pcs (1 to 4)
To ensure the high availability of proxy, please purchase at least two proxy nodes.
It's recommended to set the number of proxy nodes to 1/8 (rounded up to the nearest integer) of the sum of the CPU cores per node of the source instance and the CPU cores of all its read-only instances. For example, if the source instance uses 4 CPU cores per node and its read-only instances use 8 CPU cores in total, then the recommended number of proxy nodes is (4+8)/8 ≈ 2.
If the recommended number of proxy nodes you calculated exceeds the maximum purchasable quantity, please choose a higher proxy node specification.
Security Group
Open all ports-201909231610234984 3
Selected 1 item
Open all ports-201909231610234984 3
Preview Rules
Instruction
To access through the database proxy, you need to configure security group policies and open the private port (3306). For more information, see MySQL Security Groups
Remarks
Enter remarks.
OK
Cancel

4. , , , .

:

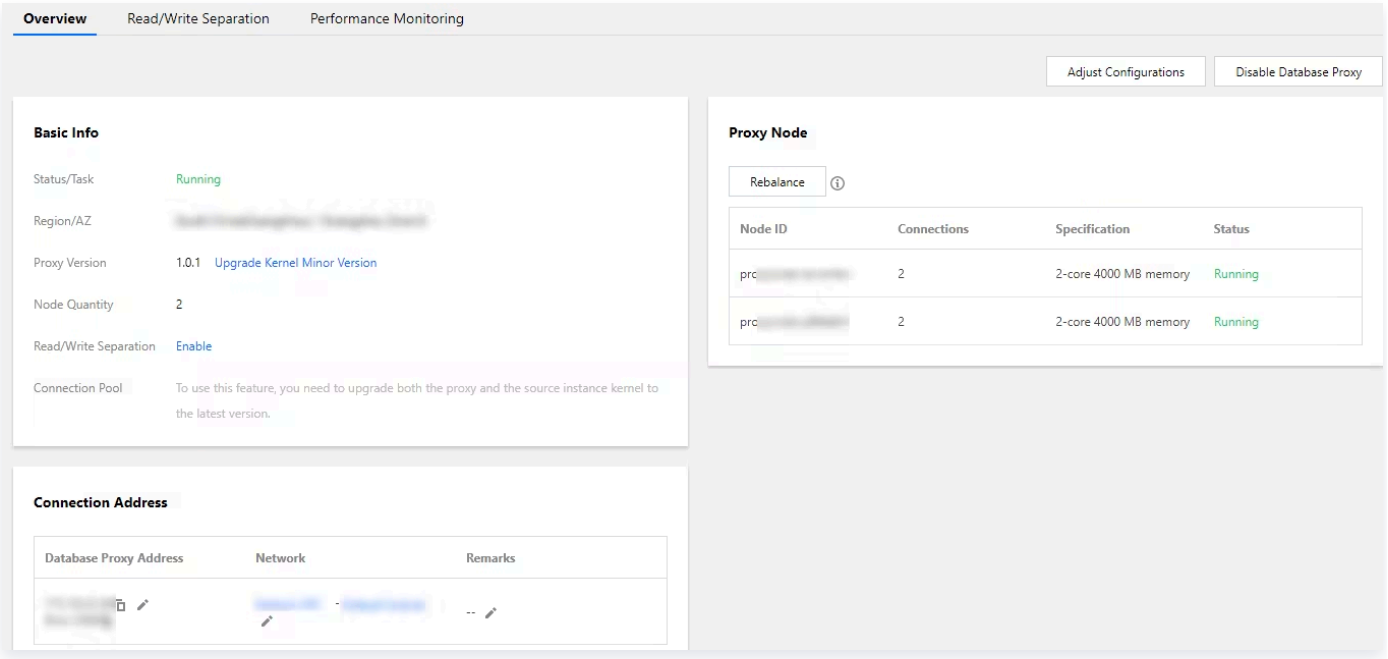
,

.

.

.

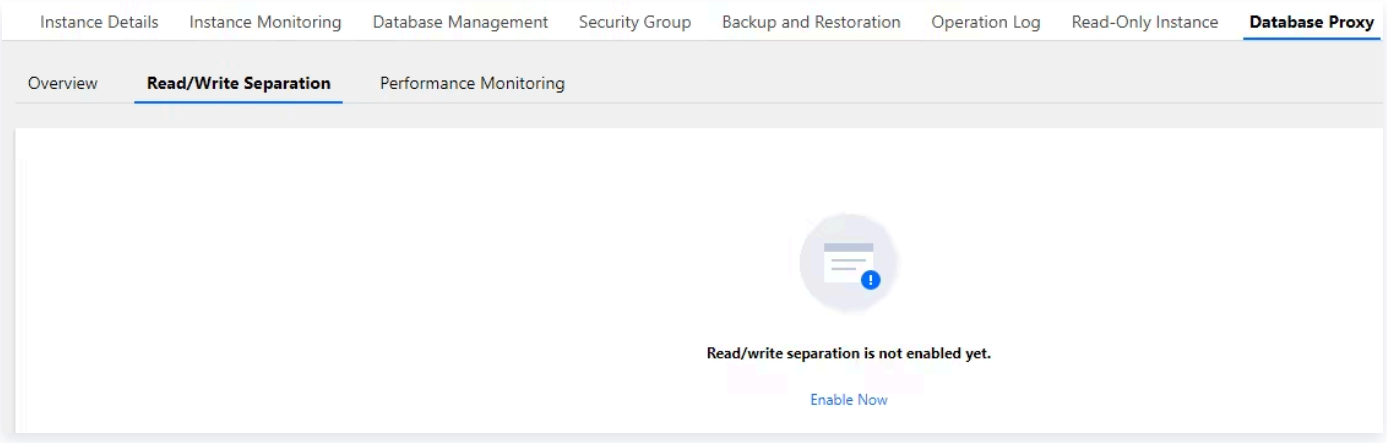
.



2 : /

1. MySQL .
2. ID .
3. /

.



4. .

: .
RO RO .

: . (,) , / . 10 , 1 .

: .

: .

1 .

, 0 (‘ , , .)

.

, .

, / .

: .

(:) , .

: , , .

: , 0 – 100 .

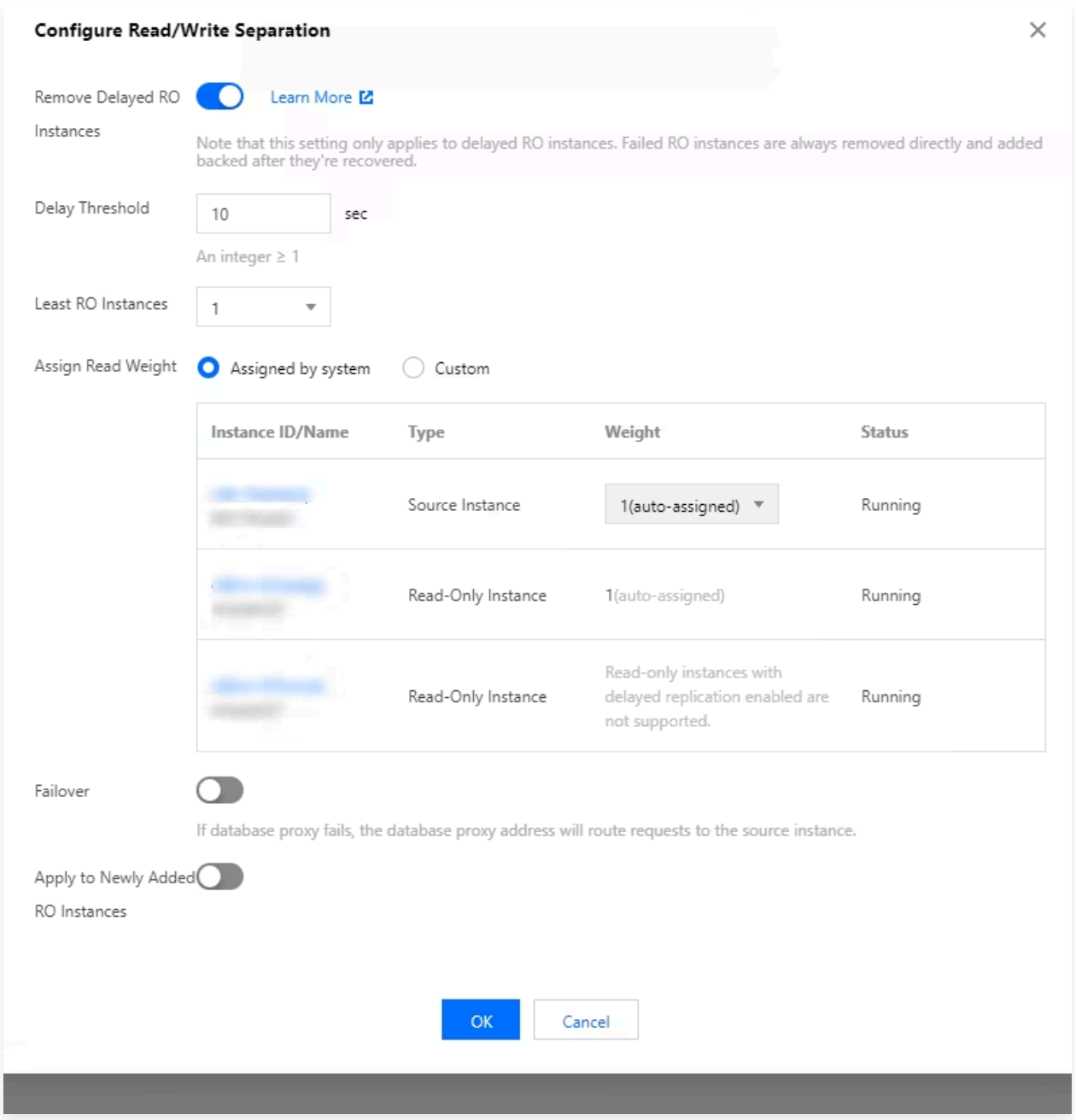
- . , 2
  10 20 , 1:2
  .
- .
  10 10
  1:1 10
  5 5 .
- CPU
  .
- 0 .
  0 .

: .
.

: .

: .
.

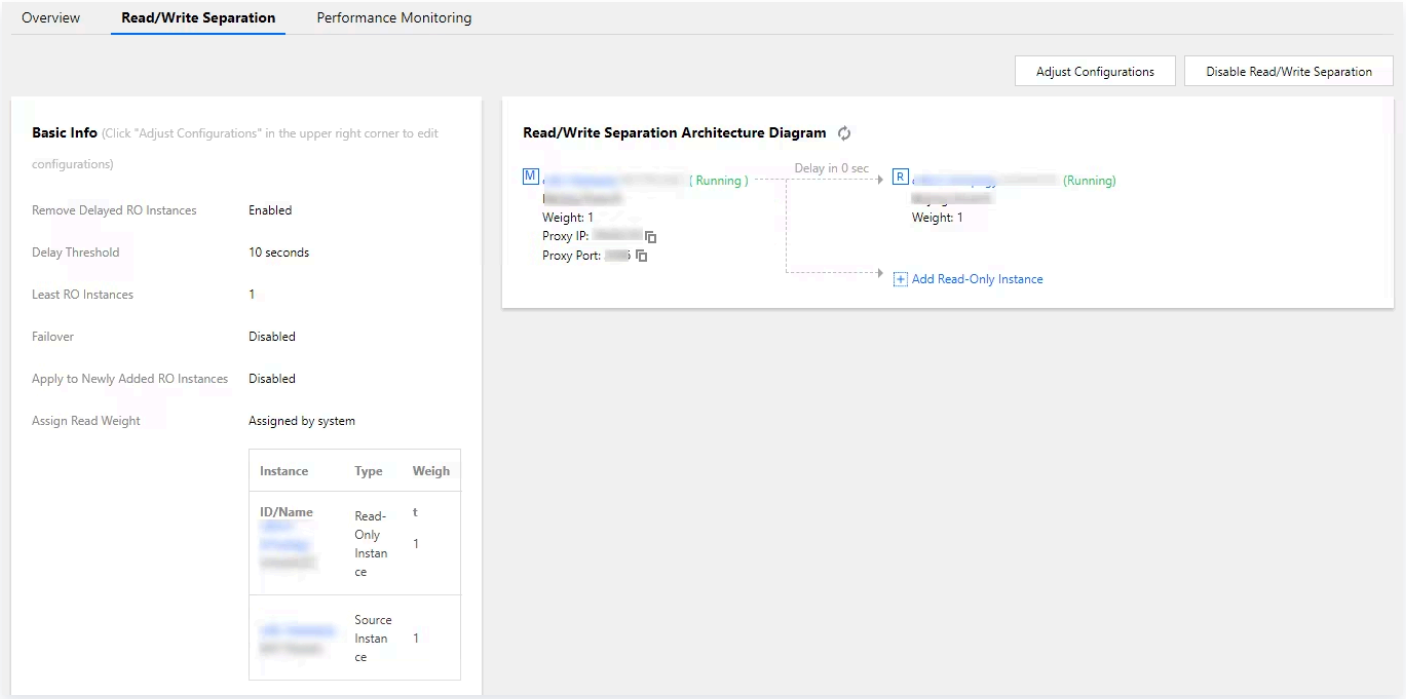
- .
- 0 ,
  / .


Configure Read/Write Separation
Remove Delayed RO Instances
Learn More
Note that this setting only applies to delayed RO instances. Failed RO instances are always removed directly and added backed after they're recovered.
Delay Threshold
10
sec
An integer ≥ 1
Least RO Instances
1
Assign Read Weight
Assigned by system
Custom
Instance ID/Name
Type
Weight
Status
Source Instance
1(auto-assigned)
Running
Read-Only Instance
1(auto-assigned)
Running
Read-Only Instance
Read-only instances with delayed replication enabled are not supported.
Running
Failover
If database proxy fails, the database proxy address will route requests to the source instance.
Apply to Newly Added RO Instances
OK
Cancel

/

/

.

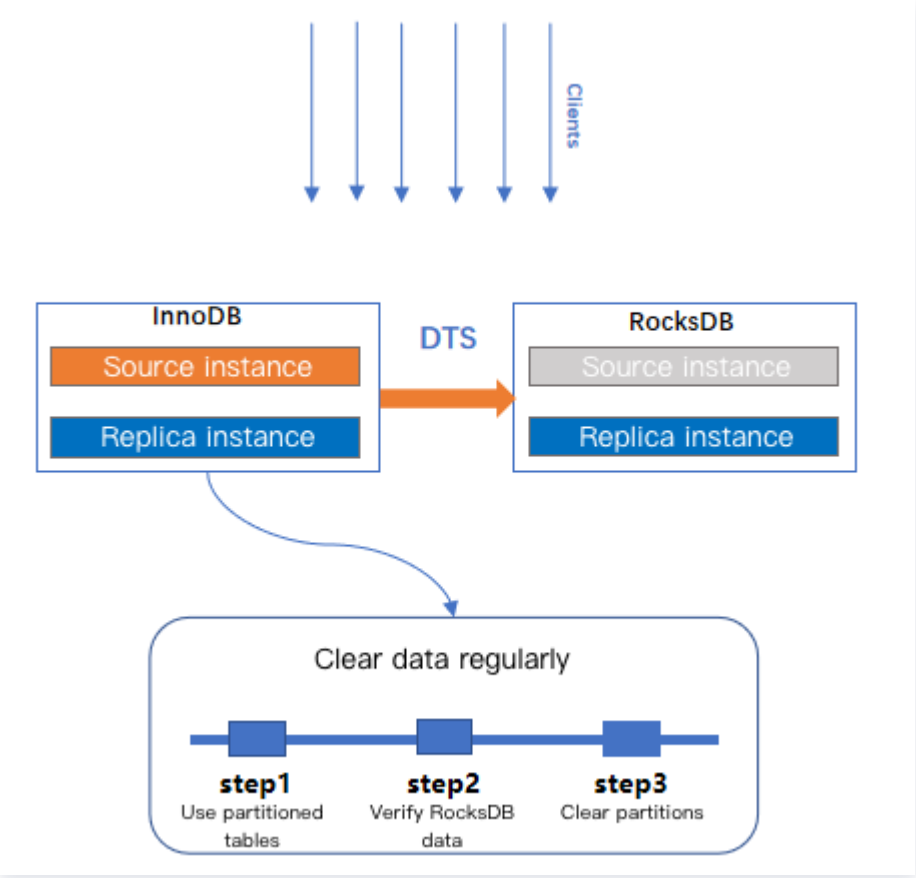

Overview
Read/Write Separation
Performance Monitoring
Adjust Configurations
Disable Read/Write Separation
Basic Info (Click "Adjust Configurations" in the upper right corner to edit configurations)
Remove Delayed RO Instances
Enabled
Delay Threshold
10 seconds
Least RO Instances
1
Failover
Disabled
Apply to Newly Added RO Instances
Disabled
Assign Read Weight
Assigned by system
Instance
Type
Weigh
ID/Name
Read-Only Instance
t
1
Source Instance
1
Read/Write Separation Architecture Diagram
{ Running }
Delay in 0 sec
(Running)
Weight: 1
Proxy IP:
Proxy Port:
Weight: 1
Add Read-Only Instance

DTS InnoDB RocksDB

:: 2024-07-25 16:38:48

TXRocks KV(key-value) RocksDB Tencent TXSQL
.

InnoDB B+Tree TXRocks LSM Tree
. InnoDB B+Tree half-full , .
InnoDB . TXRocks SST MB
. TXRocks 4K . SST Block
Block . TXRocks SST
. SST
. 50% .
InnoDB DTS RocksDB .



:
DTS(Data Transmission Service) ,

TencentDB

- DTS RocksDB Delete
- RocksDB
- (
).
- RocksDB compaction
- RocksDB Instructions

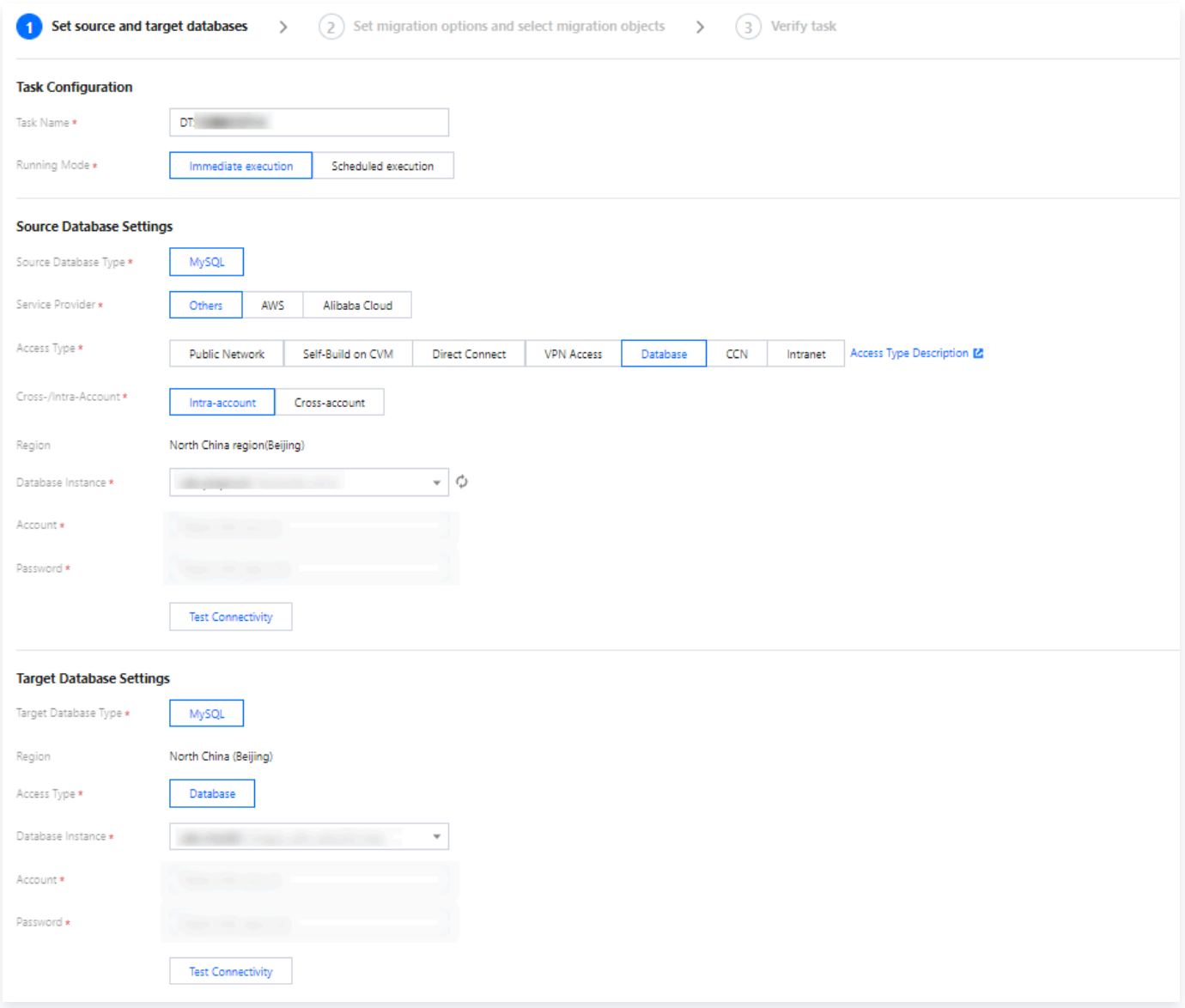
1. DTS
2. ,

	‘MySQL’
	‘MySQL’

3. >
, .

:

DB .



4.

.

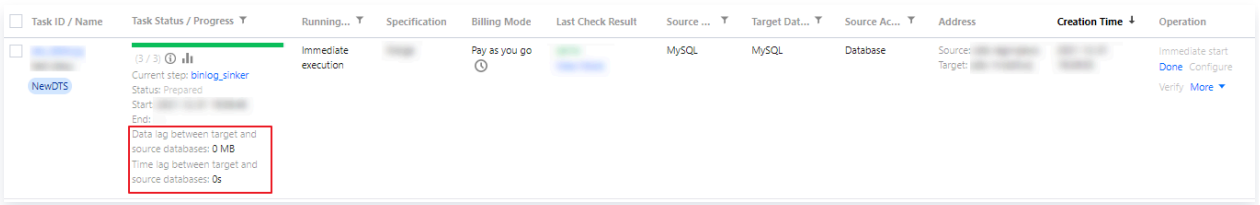
:

- Online DDL gh-ost pt-osc
,
( ) . Online DDL
.

- rename (: table A table B rename)
  table A table A (
) . .

	. • : . • : . . • + : . .
	• : information_schema, mysql, performance_schema sys . • : .
	.
	.

5. .
6. . 1 – 2
   .
   - : .
   - + :
     .
     .
     - .
     - .
       .
     - 0MB 0
       .



7. InnoDB RocksDB .

LAMP

:: 2024-07-25 16:38:48

LAMP Linux+Apache+Mysql/MariaDB+Perl/PHP/Python ,
. ,
. Web .
: 1 Tencent CDB ,
Tencent CVM 1 LAMP , Tencent CDB
.
Tencent CDB .
. , ,
.

: CDB CVM . CDB
CVM , .

CDB

CDB MySQL .

CVM .

CVM Linux CVM . CentOS
.

MySQL

1. CVM `yum` MySQL .

```
yum install mysql -y
```

```
[root@VM_165_193_centos html]# yum install mysql -y
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package mariadb.x86_64 1:5.5.52-1.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch                Version                    Reposit
================================================================================
Installing:
 mariadb                x86_64              1:5.5.52-1.el7             os

Transaction Summary
================================================================================
Install  1 Package

Total download size: 8.7 M
Installed size: 48 M
Downloading packages:
mariadb-5.5.52-1.el7.x86_64.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : 1:mariadb-5.5.52-1.el7.x86_64
  Verifying  : 1:mariadb-5.5.52-1.el7.x86_64

Installed:
  mariadb.x86_64 1:5.5.52-1.el7

Complete!
```

2. Tencent CDB .

```
mysql -h hostname -u username -p
```

```
[root@VM_165_193_centos html]# mysql -h  -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 19768
Server version: 5.6.28-cdb2016-log 20170228

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

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

MySQL [(none)]>
```

, hostname IP , username

.

3. .

```
quit;
```

Apache

1. CVM `yum` Apache .

```
yum install httpd -y
```

```
[root@VM_165_193_centos html]# yum install httpd -y
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.6-45.el7.centos.4 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                 Arch                    Version
================================================================================
Installing:
 httpd                   x86_64                  2.4.6-45.el7.centos.4

Transaction Summary
================================================================================
Install  1 Package

Total download size: 2.7 M
Installed size: 9.4 M
Downloading packages:
httpd-2.4.6-45.el7.centos.4.x86_64.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : httpd-2.4.6-45.el7.centos.4.x86_64
  Verifying  : httpd-2.4.6-45.el7.centos.4.x86_64

Installed:
  httpd.x86_64 0:2.4.6-45.el7.centos.4

Complete!
```

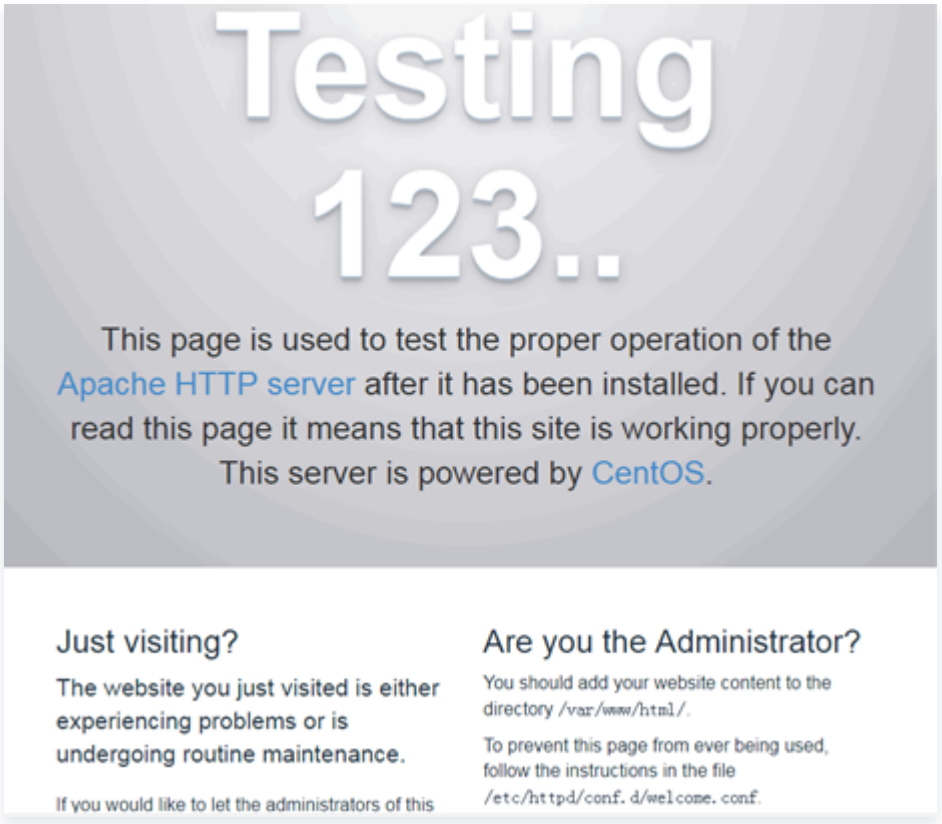
2. Apache .

```
service httpd start
```

3. Apache

> : CVM all , TCP:80 Inbound rule . Security Group .

`http://xxx.xxx.xxx.xxx/` (, `xxx.xxx.xxx.xxx` CVM IP), Apache .



PHP

1. CVM `yum` PHP .

```
yum install php -y
```

```
[root@VM_165_193_centos html]# yum install php -y
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
Resolving Dependencies
--> Running transaction check
---> Package php.x86_64 0:5.4.16-42.el7 will be installed
--> Processing Dependency: php-common(x86-64) = 5.4.16-42.el7 for p
--> Processing Dependency: php-cli(x86-64) = 5.4.16-42.el7 for pack
--> Running transaction check
---> Package php-cli.x86_64 0:5.4.16-42.el7 will be installed
---> Package php-common.x86_64 0:5.4.16-42.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                        Arch                           Versio
================================================================================
Installing:
 php                            x86_64                         5.4.16
Installing for dependencies:
 php-cli                        x86_64                         5.4.16
 php-common                     x86_64                         5.4.16

Transaction Summary
================================================================================
Install  1 Package (+2 Dependent packages)

Total download size: 4.6 M
Installed size: 17 M
Downloading packages:
(1/3): php-5.4.16-42.el7.x86_64.rpm
(2/3): php-common-5.4.16-42.el7.x86_64.rpm
(3/3): php-cli-5.4.16-42.el7.x86_64.rpm
--------------------------------------------------------------------------------
Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
```

LAMP

1. , CVM `/var/www/html` 1 info.php .

```
<?php phpinfo(); ?>
```

2. Apache .

```
service httpd restart
```

3. `http://xxx.xxx.xxx.xxx/info.php` (, `xxx.xxx.xxx.xxx` CVM IP), LAMP .

PHP Version 5.4.16

System	Linux VM_165_193_centos 3.10.0-327.36.3.el7.x86_64 #1 SMP Mon Oct 24 16:09:20 UTC 2016 x86_64
Build Date	Nov 6 2016 00:30:05
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/mysql.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/sqlite3.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend	API220100525,NTS

Drupal

:: 2024-07-25 16:38:48

Drupal PHP (Content Management Framework) , (Content Management System) PHP (Framework) . Drupal

.

CVM Drupal .

: CentOS7.2, Drupal7.56, PHP5.4.16.

CVM

CVM Linux CVM .

MariaDB

1. MariaDB CentOS v7 MariaDB . `yum` CVM MariaDB .

```
yum install mariadb-server mariadb -y
```

2. MariaDB .

```
systemctl start mariadb
```

3. Drupal (: drupal) .

```
mysqladmin -u root -p create drupal
```

drupal Drupal .

4. .

```
mysql -u root -p
```

.

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER,
CREATE TEMPORARY TABLES, LOCK TABLES ON drupal.* TO
'username'@'localhost' IDENTIFIED BY 'password';
```

```
FLUSH PRIVILEGES;
exit
```

username Drupal , password Drupal

.

Apache

1. CVM `yum` Apache .

```
yum install httpd -y
```

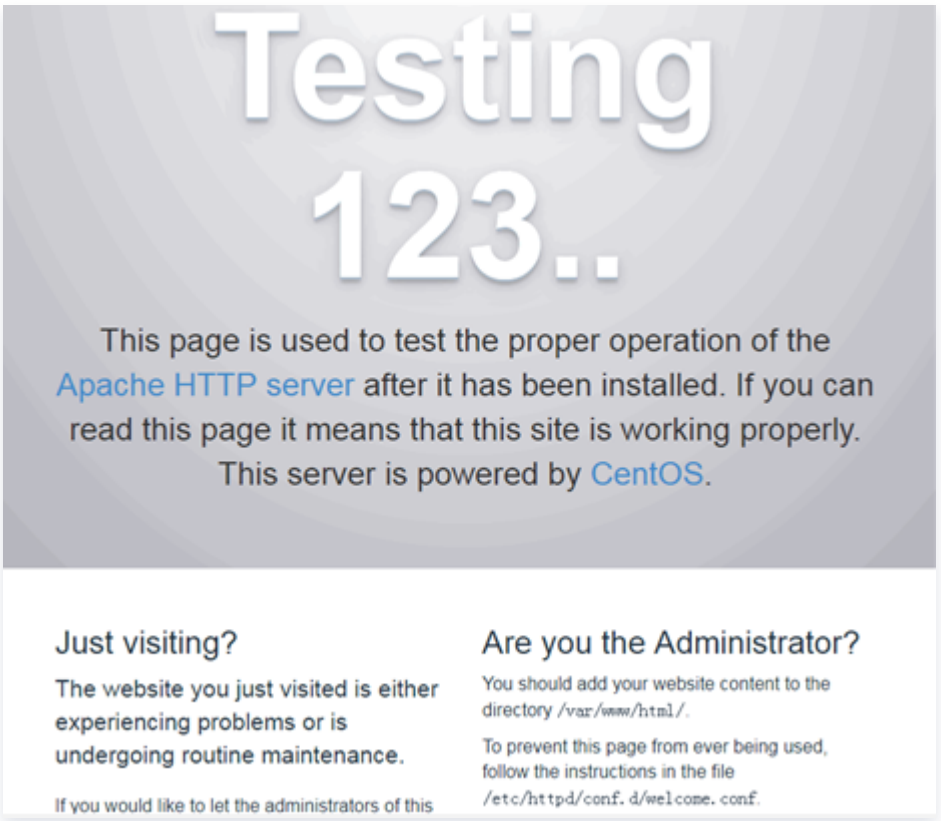
2. Apache .

```
service httpd start
```

3. Apache .

> : CVM all TCP:80
> .
> .

`http://115.xxx.xxx.xxx/` (`115.xxx.xxx.xxx` CVM IP). Apahce .



PHP

1. `yum` CVM PHP .

```
yum install php php-dom php-gd php-mysql php-pdo -y
```

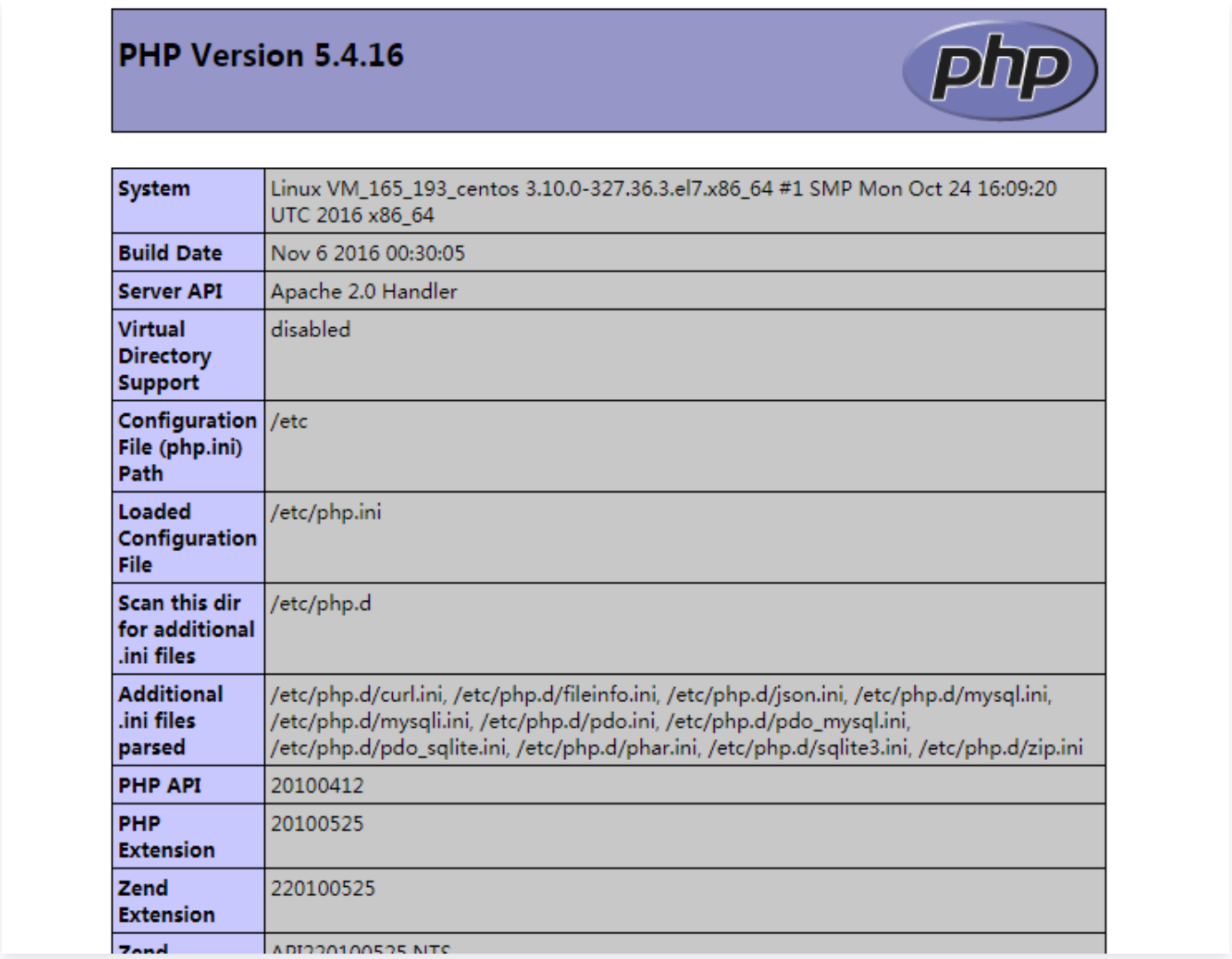
2. , CVM `/var/www/html` info.php PHP .

```
<?php phpinfo(); ?>
```

3. Apache .

```
service httpd restart
```

4. `http://115.xxx.xxx.xxx/info.php` (`115.xxx.xxx.xx x` CVM IP). PHP .



PHP Version 5.4.16

System	Linux VM_165_193_centos 3.10.0-327.36.3.el7.x86_64 #1 SMP Mon Oct 24 16:09:20 UTC 2016 x86_64
Build Date	Nov 6 2016 00:30:05
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/mysql.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/sqlite3.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend	API220100525,NTS

Drupal

1. Drupal .

```
wget http://ftp.drupal.org/files/projects/drupal-7.56.zip
```

2. .

```
unzip drupal-7.56.zip
mv drupal-7.56/* /var/www/html/
```

3. .

```
cd /var/www/html/
```

```
wget -P profiles/standard/translations
http://ftp.drupal.org/files/translations/7.x/drupal/drupal-7.56.zh-
hans.po
```

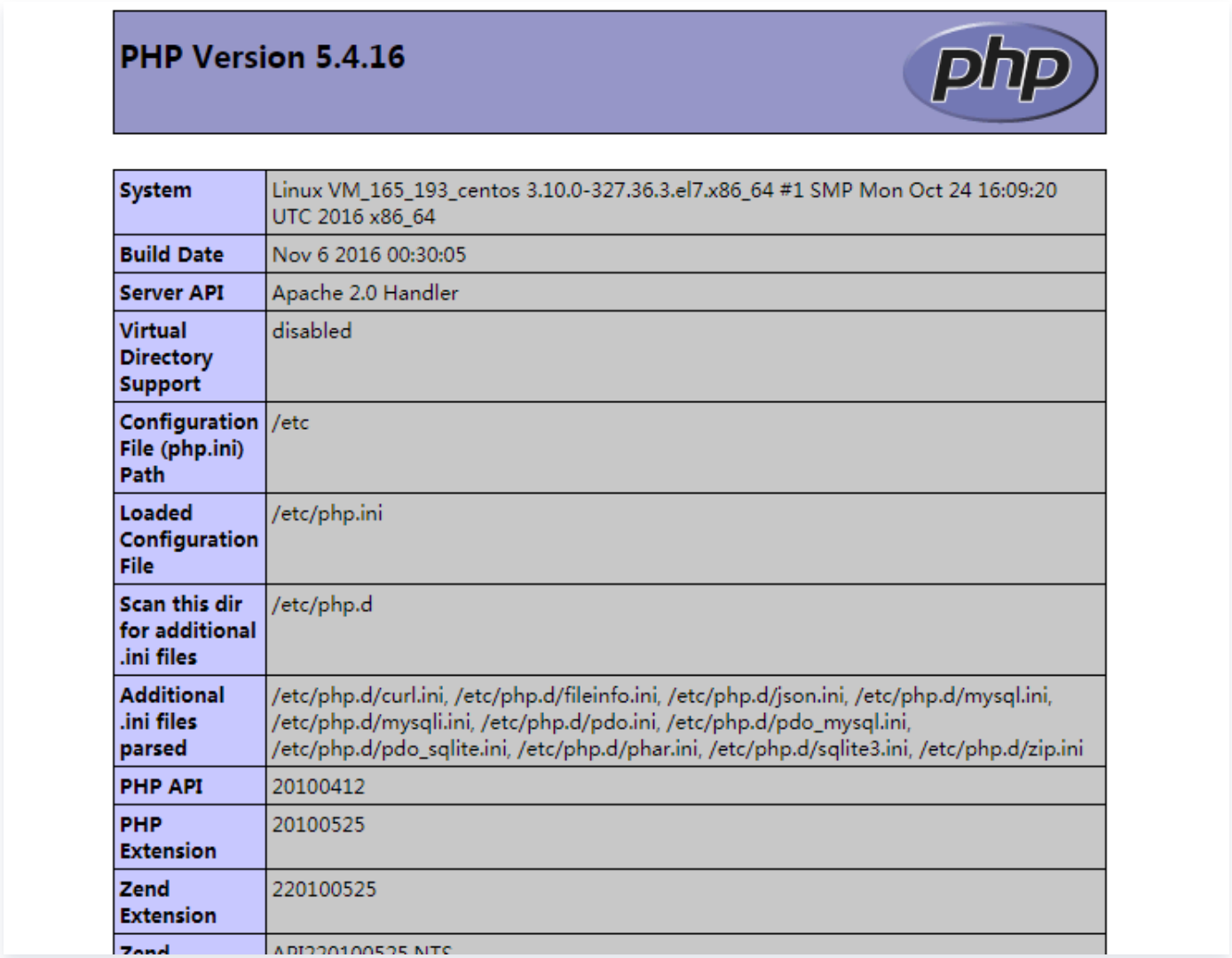
4. `sites` .

```
chown -R apache:apache /var/www/html/sites
```

5. Apache .

```
service httpd restart
```

6. `http://115.xxx.xxx.xxx/` (`115.xxx.xxx.xxx` CVM IP) Drupal [Save and continue] .



PHP Version 5.4.16

System	Linux VM_165_193_centos 3.10.0-327.36.3.el7.x86_64 #1 SMP Mon Oct 24 16:09:20 UTC 2016 x86_64
Build Date	Nov 6 2016 00:30:05
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/mysql.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/sqlite3.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend	API220100525,NTS

7. , [Save and continue] .
8. , mariadb .
9. .
10. Drupal .
11. `http://115.xxx.xxx.xxx/` (`115.xxx.xxx.xxx` CVM IP)
.

Python MySQL API

:: 2024-07-25 16:38:48

API	
CreateDBInstance	CDB
CreateDBInstanceHour	CDB ( )
DescribeDBInstances	
DescribeDBPrice	
DescribeDBZoneConfig	CDB
InitDBInstances	

CreateDBInstance CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models


'''??? ???? ??'''
def CreateDBInstancedemomaster():
    try:
        # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId,
secretKey? ?? ??
        cred = credential.Credential("secretId", "secretKey")

        #????? ? ??(?: cdb)? client ?? ?? ??
```

```
        client = cdb_client.CdbClient(cred, "ap-beijing")

        #?? ?? ??????:req = models.ModifyInstanceParamRequest()
        req = models.CreateDBInstanceRequest()
        req.Memory = 2000
        req.Volume = 120
        req.Period = 1
        req.GoodsNum =1
        req.Zone = "ap-beijing-1"
        req.Port = 3306
        #req.MasterInstanceId = "cdb-7ghaiocc"
        req.InstanceRole = "master"
        req.EngineVersion = "5.6"
        req.Password = "CDB@Qcloud"
        req.ProtectMode = 0
        req.InstanceName = "tencentcdb"
        req.SecurityGroup = ["sg-eq0hvlzp"]


        # client ??? ?? ????? ??????? ??, ?? ?? ?? ??
        resp = client.CreateDBInstance(req)
        # json ??? ???? ??
        print(resp.to_json_string())


    except TencentCloudSDKException as err:
        msg = traceback.format_exc() # ??1
        print (msg)




'''?? ?? ????? ??'''
def CreateDBInstancedemoro():
    try:
        # 1?? ?? ??? ??????, ????? ?? ? Tencent Cloud ?? secretId,
secretKey? ?? ??
        cred = credential.Credential("secretId", "secretId")

        #?????? ? ??(?: cdb)? client ?? ?? ??
```

```
        client = cdb_client.CdbClient(cred, "ap-beijing")

        #?? ?? ??????:req = models.ModifyInstanceParamRequest()
        req = models.CreateDBInstanceRequest()
        req.Memory = 2000
        req.Volume = 200
        req.Period = 1
        req.GoodsNum =1
        req.Zone = "ap-beijing-1"
        req.Port = 3306
        req.InstanceRole = "ro"
        req.EngineVersion = "5.6"
        req.Password = "CDB@Qcloud"
        req.ProtectMode = 0
        req.DeployMode =1
        req.GoodsNum =2
        req.SlaveZone = "ap-beijing-1"
        req.ParamList = [{"name":"max_connections","value":"1000"},
{"name":"lower_case_table_names","value":"1"}]
        req.BackupZone = "0"
        req.AutoRenewFlag =0
        req.MasterInstanceId ="cdb-bgr97hu0"
        req.RoGroup = {"RoGroupMode":"allinone","RoGroupName":"roweek"}
        req.InstanceName = "tencentcdbRO"


        # client ??? ?? ????? ??????? ??, ?? ?? ?? ??
        resp = client.CreateDBInstance(req)
        # json ??? ??? ??
        print(resp.to_json_string())


    except TencentCloudSDKException as err:
        msg = traceback.format_exc() # ??1
        print (msg)


'''?? ?? ???? ??'''
def CreateDBInstancedemodr():
    try:
```

```
        # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId,
secretKey에 전송 필요
        cred = credential.Credential("secretId", "secretKey")

        #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
        client = cdb_client.CdbClient(cred, "ap-shanghai")

        #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
        req = models.CreateDBInstanceRequest()

        req.Memory = 4000
        req.Volume = 200
        req.Period = 1
        req.GoodsNum =1
        #req.Zone = "ap-shanghai-2"
        req.Port = 3306
        req.InstanceRole = "dr"
        #req.MasterInstanceId
        req.EngineVersion = "5.6"
        req.Password = "CDB@Qcloud"
        req.ProtectMode = 0
        req.DeployMode =0
        #req.SlaveZone = "ap-guangzhou-3"
        req.ParamList = [{"name":"max_connections","value":"1000"},
{"name":"lower_case_table_names","value":"1"}]
        req.BackupZone = "0"
        req.AutoRenewFlag =0
        #req.RoGroup = {"RoGroupMode":"alone","RoGroupName":"roweek"}
        #req.RoGroup = {"RoGroupName":"roweek"}
        #param = models.RoGroup()
        #param.RoGroupMode = "alone"
        #param.RoGroupName = "roweek"
        #param.MinRoInGroup = 1
        #req.RoGroup = [param]


        #ro = [{"roGroupMode":"allinone"},{"RoGroupName":"ro_www"}]
        #req.RoGroup = [ro]
        req.MasterInstanceId ="cdb-bgr97hu0"
        req.MasterRegion = "ap-beijing"
```

```
        #roGroup = [RoGroupMode="allinone",
RoGroupName="weekro",RoOfflineDelay=1,MinRoInGroup=5,MinRoInGroup=1]
        #req.RoGroup = [roGroup]
        req.InstanceName = "tencentcdbDR"



        # client ??? ?? ???? ????? ??, ?? ?? ?? ??
        resp = client.CreateDBInstance(req)
        # json ??? ??? ??
        print(resp.to_json_string())


    except TencentCloudSDKException as err:
        msg = traceback.format_exc() # ??1
        print (msg)

#CreateDBInstancedemodr()
#CreateDBInstancedemoro()
#CreateDBInstancedemomaster()
```

CreateDBInstanceHour CDB ( )

```
'''??? ??? ?? ?? ???? ???? ??? ??? ???? ??? ?. ?? ??? 0? ?? ?? ??'''
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")
```

```
    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-beijing")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.CreateDBInstanceHourRequest()
    req.EngineVersion = "5.6"
    req.Zone = "ap-beijing-3"
    req.ProjectId = 0
    req.GoodsNum = 1
    req.Memory = 1000
    req.Volume = 50
    req.InstanceRole = "master"
    req.Port =3311
    req.Password = "CDB@Qcloud"
    req.ParamList = [{"name":"max_connections","value":"1000"},
{"name":"lower_case_table_names","value":"1"}]
    req.ProtectMode = 1
    req.SlaveZone = "ap-beijing-3"
    req.InstanceName = "oneday1"
    req.AutoRenewFlag = 0


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.CreateDBInstanceHour(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    msg = traceback.format_exc() # 방식1
    print (msg)
```

DescribeDBInstances

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기
import logging
```

```
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstancesRequest()
    req.EngineVersions = ["5.6"]
    req.OrderBy = "instanceId"
    req.InstanceIds = ["cdb-1j8lumf6"]


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeDBInstances(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
        msg = traceback.format_exc() # 방식1
        print (msg)
```

DescribeDBPrice

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
```

```
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-guangzhou")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBPriceRequest()
    req.Zone = "ap-guangzhou-3"
    req.GoodsNum = 1
    req.Memory =2000
    req.Volume =1000
    req.PayType = 'PRE_PAID'
    req.Period=1


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeDBPrice(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeDBZoneConfig CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
```

```
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBZoneConfigRequest()
    #req.InstanceId = "cdb-j0edpju5"


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeDBZoneConfig(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

InitDBInstances

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
```

```
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.InitDBInstancesRequest()
    req.InstanceIds = ["cdb-c752yqcn"]
    req.NewPassword = "CDB@Qcloud"

    req.Parameters = [{"name":"max_connections","value":"100"},
{"name":"character_set_server","value":"utf8"},
{"name":"lower_case_table_names","value":"1"}]


    # client 객체를 통해 액세□할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.InitDBInstances(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

:: 2024-07-25 16:38:48

API	
ModifyInstanceParam	
CloseWanService	
OpenWanService	
RestartDBInstances	
OpenDBInstanceGTID	GTID
ModifyDBInstanceName	CDB
ModifyDBInstanceProject	CDB
ModifyDBInstanceVipVport	CDB IP
DescribeDBInstanceCharset	CDB
DescribeDBInstanceConfig	CDB
DescribeDBInstanceGTID	CDB GTID
DescribeDBInstanceRebootTime	CDB

ModifyInstanceParam

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# [illegible] API [illegible] [illegible] [illegible]
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
```

```
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화
    req = models.ModifyInstanceParamRequest()
    req.InstanceIds = ["cdb-1y6g3zj8","cdb-7ghaiocc"]
    req.ParamList = [{"name":"max_connections","currentValue":"100"},
{"name":"character_set_server","currentValue":"utf8"},
{"name":"lower_case_table_names","currentValue":"1"}]
    #req.ParamList = [{"name":"max_connections","currentValue":"100"}]
    #param = models.Parameter()
    #param.Name = "max_connections"
    #param.CurrentValue = "1000"
    #req.ParamList = [param]



    print req
    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.ModifyInstanceParam(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    msg = traceback.format_exc() # 방식1
    print (msg)
```

CloseWanService

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
```

```
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.CloseWanServiceRequest()
    req.InstanceId = "cdb-1y6g3zj8"

    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.CloseWanService(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

OpenWanService

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
```

```
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.OpenWanServiceRequest()
    req.InstanceId = "cdb-1y6g3zj8"

    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.OpenWanService(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

RestartDBInstances

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")
```

```
    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstancesRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"

    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.InitDBInstances(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

OpenDBInstanceGTID GTID

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.OpenDBInstanceGTIDRequest()
    req.InstanceId = "cdb-7ghaiocc"


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.OpenDBInstanceGTID(req)
```

```
    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

ModifyDBInstanceName CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-beijing")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.ModifyDBInstanceNameRequest()
    req.InstanceId = "cdb-cukm86n2"
    req.InstanceName = "1s ???"

    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.ModifyDBInstanceName(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

ModifyDBInstanceProject CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models


def DescribeDBInstancesList():
    try:
        # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId,
secretKey? ?? ??
        cred = credential.Credential("secretId", "secretKey")

        #????? ? ??(?: cdb)? client ?? ?? ??
        client = cdb_client.CdbClient(cred, "ap-shanghai")

        #?? ?? ?????:req = models.ModifyInstanceParamRequest()
        req = models.ModifyDBInstanceProjectRequest()
        req.InstanceIds = ["cdb-7ghaiocc"]
        req.NewProjectId =1


        # client ??? ?? ???? ?????? ??, ?? ?? ?? ??
        resp = client.ModifyDBInstanceProject(req)

        # json ??? ??? ??
        print(resp.to_json_string())
    except TencentCloudSDKException as err:
        msg = traceback.format_exc() # ??1
        print (msg)
```

```
DescribeDBInstancesList()
```

ModifyDBInstanceVipVport CDB IP

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.ModifyDBInstanceVipVportRequest()
    req.InstanceId = "cdb-7ghaiocc"
    req.DstIp = "10.0.0.13"
    req.DstPort =1025
    req.UniqVpcId = 1111

    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.ModifyDBInstanceVipVport(req)

    # json ??? ??? ??
```

```
    print(resp.to_json_string())
except TencentCloudSDKException as err:
        msg = traceback.format_exc() # ??1
        print (msg)
```

DescribeDBInstanceCharset CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstanceCharsetRequest()
    req.InstanceId = "cdb-1y6g3zj8"


    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.DescribeDBInstanceCharset(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeDBInstanceConfig CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstanceConfigRequest()
    req.InstanceId = "cdb-1y6g3zj8"


    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.DescribeDBInstanceConfig(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeDBInstanceGTID CDB GTID

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
```

```
# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstanceGTIDRequest()
    req.InstanceId = "cdb-1y6g3zj8"


    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.DescribeDBInstanceGTID(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeDBInstanceRebootTime<42/>CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
```

```
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBInstanceRebootTimeRequest()
    req.InstanceIds = ["cdb-1y6g3zj8"]


    # client 객체를 통해 액세□할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeDBInstanceRebootTime(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

:: 2024-07-25 16:38:48

API	
CreateBackup	CDB
DeleteBackup	CDB
DescribeBackupConfig	CDB
DescribeBackupDatabases	
DescribeBackupTables	
DescribeBackups	
DescribeBinlogs	
DescribeSlowLogs	
ModifyBackupConfig	

CreateBackup CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")
```

```
    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.CreateBackupRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"
    req.BackupMethod = "logical"

    print req
    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.CreateBackup(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DeleteBackup CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")



    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DeleteBackupRequest()
```

```
    #req.MasterInstanceId = "cdb-7ghaiocc"
    #print  req.BackupId
    req.BackupId = 105119782


    # client ??? ?? ????? ?????? ??, ?? ?? ?? ??
    resp = client.DeleteBackup(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeBackupConfig CDB

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBZoneConfigRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"

    print req
```

```
    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeDBZoneConfig(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeBackupDatabases

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기
import logging
import traceback
from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeBackupDatabasesRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"
    req.StartTime = "2018-08-02 15:19:19"

    print req
    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeBackupDatabases(req)

    # json 포맷의 문자열 출력
```

```
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    msg = traceback.format_exc() # ??1
    print (msg)
```

DescribeBackupTables

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeBackupTablesRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"
    req.StartTime = "2018-08-02 15:19:19"
    req.DatabaseName ="sissi"


    # client ??? ?? ???? ?????? ??, ?? ?? ?? ??
    resp = client.DescribeBackupTables(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeBackups

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBPriceRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"



    # client ??? ?? ???? ????? ??, ?? ?? ?? ??
    resp = client.DescribeBackups(req)
    print resp

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeBinlogs

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
```

```
# ???? API ??? ?? ????

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1?? ?? ??? ?????, ???? ?? ? Tencent Cloud ?? secretId, secretKey?
?? ??
    cred = credential.Credential("secretId", "secretKey")

    #?????? ? ??(?: cdb)? client ?? ?? ??
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #?? ?? ?????:req = models.ModifyInstanceParamRequest()
    req = models.DescribeDBPriceRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"


    # client ??? ?? ???? ?????? ??, ?? ?? ?? ??
    resp = client.DescribeDBInstances(req)

    # json ??? ??? ??
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

DescribeSlowLogs

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# ???? API ??? ?? ????

from tencentcloud.common import credential
```

```
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.DescribeSlowLogsRequest()
    #req.MasterInstanceId = "cdb-7ghaiocc"


    # client 객체를 통해 액세스할 인터페이스 호출, 요청 객체 전송 필요
    resp = client.DescribeSlowLogs(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```

ModifyBackupConfig

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

# 클라우드 API 게이트 모듈 불러오기

from tencentcloud.common import credential
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.cdb.v20170320 import cdb_client, models

try:
```

```
    # 1개의 인증 객체를 인스턴스화, 매개변수 입력 시 Tencent Cloud 계정 secretId, secretKey에
전송 필요
    cred = credential.Credential("secretId", "secretKey")

    #인스턴스화 시 제품(예: cdb)의 client 객체 요청 필요
    client = cdb_client.CdbClient(cred, "ap-shanghai")

    #요청 객체 인스턴스화:req = models.ModifyInstanceParamRequest()
    req = models.ModifyBackupConfigRequest()
    req.InstanceId = "cdb-1y6g3zj8"
    req.ExpireDays = 10
    req.StartTime = "06:00-10:00"
    req.BackupMethod = "logical"
    print req


    # client 객□를 통해 □세□할 □터페이스 호□, 요청 객체 전송 필□
    resp = client.ModifyBackupConfig(req)

    # json 포맷의 문자열 출력
    print(resp.to_json_string())
except TencentCloudSDKException as err:
    print(err)
```