

天翼云关系型数据库

故障处理

目 录

1 怎么解决查询运行缓慢的问题	4
2 查看 RDS 存储空间使用量	5
3 RDS 主备复制关系异常	6
4 MySQL 实例无法访问	7
5 MySQL 大小写参数敏感类问题	11
6 MySQL 的查询缓存功能何时该开启	12
7 使用 Mysqldump 导出大表的注意事项	14
8 MySQL 客户端连接实例后会自动断开	15
9 Mysqldump 的 6 大使用场景的导出命令	17
10 从阿里云迁移至天翼云的 Microsoft SQL Server 数据库无法创建用户	19
11 MySQL 增加表字段后出现运行卡顿现象	21
12 RDS 实例恢复到指定时间点失败	22
13 MySQL 数据库报错 Native error 1461 的解决方案	23
14 MySQL 创建表失败报错 Row size too large 的解决方案	24
15 MySQL 升级版本后，导致现有配置无法正常连接到 MySQL-server	25
16 磁盘空间满导致自动增量备份失败	28
17 下载备份文件在本地恢复，登录密码错误	29

A 修订记录.....	30
-------------	----

1 怎么解决查询运行缓慢的问题

1. 通过查看慢 SQL 日志来确定是否存在运行缓慢的 SQL 查询以及各个查询的性能特征（如果有），从而定位查询运行缓慢的原因。
查询 MySQL 日志，请参见[慢查询日志](#)。
查询 PostgreSQL 日志，请参见[查看错误日志](#)。
Microsoft SQL Server 可以通过查询 DMV 视图，从而定位查询运行缓慢的原因，有关使用 DMV 的信息，请参见[官网信息](#)。
2. 查看天翼云关系型数据库实例的 CPU 使用率指标，协助定位问题。
请参见[通过 Cloud Eye 监控](#)。
3. 创建只读实例专门负责查询。减轻主实例负载，分担数据库压力。
4. 多表关联查询时，关联字段要加上索引。
5. 尽量避免用 select* 语句进行全表扫描，可以指定字段或者添加 where 条件。

2 查看 RDS 存储空间使用量

场景描述

天翼云关系型数据库实例的存储空间为客户购买的数据盘存储，不包括客户后台弹性云服务器的系统盘。云监控服务（Cloud Eye）目前可以对客户存储空间的大小、使用量、利用率等作出监控及设置告警策略，但不能查看天翼云关系型数据库实例的存储空间使用量。

解决方案

步骤 1 登录管理控制台。

步骤 2 单击管理控制台左上角的 ，选择区域和项目。

步骤 3 选择“数据库 > 云数据库 RDS”。进入云数据库 RDS 信息页面。

步骤 4 在“实例管理”页面，选择指定的实例，单击实例名称。

步骤 5 进入“基本信息”页面。在“存储/备份空间”模块查看存储空间类型和使用情况。

---结束

3

RDS 主备复制关系异常

操作场景

有时候客户会遇到天翼云关系型数据库主备复制关系异常的情况，可能原因是误删除默认安全组策略，下面主要针对这个场景进行分析，供您参考。

解决方案

步骤 1 登录管理控制台。

步骤 2 单击管理控制台左上角的 ，选择区域和项目。

步骤 3 选择“数据库 > 云数据库 RDS”。进入云数据库 RDS 信息页面。

步骤 4 在“实例管理”页面，选择指定的实例，单击实例名称。

步骤 5 在“基本信息”页面，单击目标内网安全组名称，进入实例安全组页面。

步骤 6 单击“添加规则”，选择入方向，Any 协议，源地址为自身安全组，即安全组的远端要有安全组自己。

步骤 7 添加完策略之后，主备复制的关系便会恢复正常。

---结束

4 MySQL 实例无法访问

故障描述

客户端无法连接数据库，连接数据库时返回如下报错信息：

- 故障一

```
ERROR 1045 (28000): Access denied for user 'root'@'192.168.0.30' (using password:YES)
```

- 故障二

```
ERROR 1226 (42000): User 'test' has exceeded the 'max_user_connections' resource (current value:10)
```

- 故障三

```
ERROR 1129 (HY000): Host '192.168.0.111' is blocked because of many connection errors; unblock with 'mysqladmin flush-hosts'
```

故障一

步骤 1 排查密码 root 账号的密码是否正确。

一般情况下，ERROR 1045 报错为密码错误引起的，因此需要首要排除是否密码错误问题。

```
select password( 'Test1i@123');  
select host,user>Password from mysql.user where user= 'test1';
```

```
mysql> select host,user,Password from mysql.user where user='test1';
+-----+-----+-----+
| host      | user  | Password |
+-----+-----+-----+
| 192.168.0.74 | test1 | *094C0901889F4D899028A83C061EF44272590E22 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select password('testli@123');
+-----+
| password('testli@123') |
+-----+
| *094C0901889F4D899028A83C061EF44272590E22 |
+-----+
1 row in set (0.01 sec)
```

使用错误的密码登录就会失败。

```
[root@ecs-lwt-0921 ~]# mysql -utest1 -ptestli@321 -P 8635 -h192.168.0.73
Warning: Using a password on the command line interface can be insecure.
ERROR 1045 (28000): Access denied for user 'test1'@'192.168.0.74' (using password: YES)
```

步骤 2 确认该主机是否有连接数据库实例的权限。

```
select user, host from mysql.user where user='username';
```

```
mysql> select user,host from mysql.user where user='test1';
+-----+-----+
| user  | host      |
+-----+-----+
| test1 | 192.168.0.74 |
+-----+-----+
1 row in set (0.00 sec)
```

如果该数据库用户需要从其他主机登录，则需要使用 root 用户连接数据库，并给该用户授权。

以加入主机 IP 为 192.168.0.76 举例：

GRANT all privileges ON test.* TO 'test1'@'192.168.0.76' identified by 'Testli@123';

flush privileges;

```
[root@ecs-lwt-0921 ~]# mysql -uroot -ptestli@123 -P 8635 -h192.168.0.73
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 6961
Server version: 5.6.35-Log MySQL Community Server (GPL)

Copyright (c) 2000, 2016, 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> GRANT all privileges ON test.* TO 'test1'@'192.168.0.76' identified by 'testli@123';
Query OK, 0 rows affected (0.01 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.03 sec)

mysql>
```

步骤 3 确认 MySQL 客户端和实例 VIP 的连通性。

尝试进行 ping 连接性能，若可以 ping 通，排除 telnet 数据库端口的问题。

步骤 4 查看实例安全组，排查是否因安全策略问题引起的报错。

步骤 5 查询 user 表信息，确认用户信息。

```
mysql> select user,host ,max_connections,max_user_connections,password_expired from mysql.user order by user;
+-----+-----+-----+-----+-----+
| user      | host      | max_connections | max_user_connections | password_expired |
+-----+-----+-----+-----+-----+
| mysql.sys | localhost | 0               | 0                   | N               |
| rdsAdmin  | localhost | 100000          | 100000              | N               |
| rdsBackup | localhost | 100000          | 100000              | N               |
| rdsMetric | localhost | 100000          | 100000              | N               |
| rdsRepl   | 172.16.%  | 100000          | 100000              | N               |
| root      | %         | 0               | 0                   | N               |
| root      | 192.168.% | 0               | 0                   | N               |
+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

在排查中发现存在两个 root 用户。

如果用户的客户端处于 192.168 的网段，MySQL 数据库的是对 root@'192.168.%'这个用户进行认证的。而用户登录时使用的为 root@'%'这个帐号所对应的密码，因而导致连接失败，无法正常访问。此次问题是因密码错误引起的访问失败。

说明

在此案例中，root@'%'为 console 创建实例时设置密码的帐号。

---结束

故障二

步骤 1 排查是否在创建 MySQL 用户时，添加了 max_user_connections 选项，导致限制了连接数。

```
select user,host ,max_user_connections from mysql.user where user='test';
```

```
mysql> select user,host,max_user_connections from mysql.user where user='test';
+-----+-----+-----+
| user | host      | max_user_connections |
+-----+-----+-----+
| test | 192.168.0.100 | 10                   |
+-----+-----+-----+
1 row in set (0.00 sec)
```

经排查发现由于设置了 max_user_connections 选项，导致连接失败。

步骤 2 增加该用户最大连接数。

```
alter user test@ '192.168.0.100' with max_user_connections 15.
```

步骤 3 查询变更结果，检查是否可正常访问数据库。

```
mysql> alter user test@'192.168.0.100' with max_user_connections 15;
Query OK, 0 rows affected (0.01 sec)

mysql> select user,host,max_user_connections from mysql.user where user='test';
+-----+-----+-----+
| user | host          | max_user_connections |
+-----+-----+-----+
| test | 192.168.0.100 | 15                    |
+-----+-----+-----+
1 row in set (0.00 sec)
```

---结束

故障三

步骤 1 排查是否由于 MySQL 客户端连接数据库的失败次数(不包括密码错误), 超过了 max_connection_errors 的值。

步骤 2 在数据库端解除超出限值问题。使用 root 用户登录 mysql, 执行 flush hosts。
或者执行如下命令。

```
mysqladmin flush-hosts -u<user> -p<password> -h<ip> -P<port >。
```

步骤 3 再次连接, 检查是否可正常访问数据库。

---结束

5 MySQL 大小写参数敏感类问题

故障描述

用户将 MySQL 的“lower_case_table_names”设置成“大小写敏感”的状态时，创建了带有大写字母的表，如“tbl_newsTalking”，但后期改变了大小写敏感的设置状态后，无法找到该表。

案例：在执行备份恢复到新实例的时候，如果新实例的“大小写敏感”参数值与备份时原实例的参数值不一致，会导致恢复失败。

更多敏感参数，请参见《关系型数据库用户指南》中“[MySQL 参数调优建议](#)”的内容。

解决方案

- 步骤 1 修改“lower_case_table_names”值为“0”，即区分大小写。
- 步骤 2 重启数据库。
- 步骤 3 将带大写字母的表名，改为小写字母。
- 步骤 4 修改“lower_case_table_names”值为“1”，即不区分大小写。
- 步骤 5 重启数据库，或者整库导出再导入。

----结束

📖 说明

- 数据库名、表名、变量名及表的别名严格区分大小写的。
- 列名与列的别名在所有的情况下均是忽略大小写的；字段值默认是忽略大小写的。

6

MySQL 的查询缓存功能何时该开启

背景描述

查询缓存 (Query Cache, 简称 QC), 存储 SELECT 语句及其产生的数据结果。

适用场景

用于频繁提交同一个语句, 并且该表数据变化不是很频繁的场景, 例如一些静态页面, 或者页面中的某块不经常发生变化的信息。

由于查询缓存需要缓存最新数据结果, 因此表数据发生任何变化 (insert、update、delete 或其他可能产生数据变化的操作), 都会导致查询缓存被刷新。因而, 对于一个更新频率非常低而只读查询频率非常高的场景下, 打开查询缓存还是比较有优势的。

不适用场景

查询缓存严格要求 2 次 SQL 请求要完全一样, 包括 SQL 语句, 连接的数据库、协议版本、字符集等因素都会影响。下面为查询缓存不适用的几个场景:

1. 子查询;
2. 过程、函数、触发器、event 中调用的 SQL, 或者引用到这些结果的;
3. 查询中涉及一些特殊函数时, 例如: BENCHMARK()、CURDATE()、CURRENT_TIME()、CURRENT_TIMESTAMP()、NOW()、SLEEP()、CONNECTION_ID()、CURRENT_DATE()、CURRENT_USER()、PASSWORD()、RAND()、UUID()、ENCRYPT()、LAST_INSERT_ID()等等;
4. 查询涉及到 mysql, information_schema 或 performance_schema。
5. 类似 SELECT...LOCK IN SHARE MODE、SELECT...FOR UPDATE、SELECT..INTO OUTFILE/DUMPFIL、SELECT * FROM ... WHERE autoincrement_col IS NULL 的查询;
6. SELECT 执行计划用到临时表;
7. 未引用任何表的查询, 例如 SELECT 1+2;
8. 查询产生了告警(warnings);
9. SELECT 语句中存在 SQL_NO_CACHE 关键字;

10. 涉及到分区表。

可见，使用查询的缓存的限制非常多。当使用场景中以只读为主，很少有更新的情况时，再考虑[开启查询缓存](#)。

开启查询缓存

设置选项 `query_cache_type = 1`，同时设置 `query_cache_size = 67108864`;

注：`query_cache_size` 的值设置在 100MB 以内即可。在 MySQL 里查询缓存是由一个全局锁在控制，每次更新查询缓存的内存块都需要进行锁定。

关闭查询缓存

设置选项 `query_cache_type = 0`，同时设置 `query_cache_size = 0`。

7 使用 Mysqldump 导出大表的注意事项

在使用 `mysqldump` 导出数据时，倘若添加 `-q(--quick)` 参数时，`select` 出来的结果将不会存放在缓存中，而是直接导出到标准输出中。如果不添加该参数，则会把 `select` 的结果放在本地缓存中，然后再输出给客户端。

- 如果只是备份少量数据，足以放在空闲内存 `buffer` 中的话，禁用 `-q` 参数，则导出速度会快一些。
- 对于大数据集，如果没办法完全储存在内存缓存中时，就会产生 `swap`。对于大数据集的导出，不添加 `-q` 参数，不但会消耗主机的内存，也可能造成数据库主机因无可用内存继而宕机的严重后果。

因此，如果使用 `mysqldump` 来备份数据时，建议添加 `-q` 参数。

导出示例：

```
mysqldump -uroot -p-P8635 -h192.168.0.199 --set-gtid-purged=OFF --single-transaction --flush-logs -q test t1>t1.sql
```

8

MySQL 客户端连接实例后会自动断开

故障描述

MySQL 客户端连接实例后，会自动断开，报错信息：“ERROR 2013: Lost connection to MySQL server during query”。

解决方案

ERROR 2013 是 MySQL 常见错误，一般为配置错误导致。

- “wait_timeout”：服务器关闭非交互连接之前等待活动的秒数。
- “interactive_timeout”：服务器关闭交互连接之前等待活动的秒数。

步骤 1 查看实例状态是否处于正常状态。

经查看实例状态正常，继续排查其他问题。

步骤 2 查看错误日志。

步骤 3 使用 MySQL 命令行客户端连接数据库，执行 status 命令，确认数据库实例是否重频繁重启。

```
mysql> status
-----
mysql Ver 14.14 Distrib 5.6.34, for Linux (x86_64) using EditLine wrapper

Connection id:          16288
Current database:
Current user:           root@192.168.0.5
SSL:                    Not in use
Current pager:         stdout
Using outfile:         ''
Using delimiter:       ;
Server version:        5.6.34-log MySQL Community Server (GPL)
Protocol version:      10
Connection:            192.168.0.24 via TCP/IP
Server characterset:   utf8
Db characterset:       utf8
Client characterset:   utf8
Conn. characterset:    utf8
TCP port:              8635
Uptime:                5 hours 5 min 34 sec

Threads: 2 Questions: 62118 Slow queries: 0 Opens: 70 Flush tables: 2 Open tables: 0 Queries per second avg: 3.388
-----
```

Uptime 代表实例的运行时间，从排查结果可知，数据库并没有频繁重启，因而，客户端连接被断开，不是因数据库重启引起的。

- 步骤 4** 查看 “wait_timeout” 和 “interactive_timeout” 参数设置，当数值设置过小时，MySQL 会自动断开超时的空连接。
- 步骤 5** 您可根据实际应用需求量，修改 “wait_timeout” 和 “interactive_timeout” 参数值，无需重启实例。
- 步骤 6** 恢复结果确认，等到 10 分钟左右，再次执行 show databases 命令，确认连接是否正常。

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

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
+-----+
3 rows in set (0.00 sec)

mysql>
```

如图所示，说明连接正常。

----结束

9

Mysqldump 的 6 大使用场景的导出命令

背景描述

Mysqldump 是 MySQL 最常用的逻辑导入导出的工具，下面介绍几种常见使用场景。

Mysqldump 选项解析

表 9-1

选项名称	说明
add-drop-table	每个数据表创建之前添加 drop 数据表语句。
events, E	导出事件。
routines, R	存储过程以及自定义函数。
flush-logs	开始导出之前刷新日志。
no-create-db, n	只导出数据，而不添加 CREATE DATABASE 语句。
add-drop-database	创建数据库之前添加 drop 数据库语句。
no-create-info, t	只导出数据，而不添加 CREATE TABLE 语句。
no-data, d	不导出任何数据，只导出数据库表结构。
set-gtid-purged=OFF	不导出 gtid 相关语句。
hex-blob	使用十六进制格式导出二进制字符串字段。

场景描述

适用场景举例如下。

1. 导出 db1、db2 两个数据库的所有数据。

```
mysqldump -uroot -p -P8635 -h192.168.0.199 --hex-blob --set-gtid-purged=OFF  
--single-transaction --order-by-primary --flush-logs -q --databases db1  
db2 >db12.sql
```

2. 导出 db1 库的 t1 和 t2 表。

```
mysqldump -uroot -p -P8635 -h192.168.0.199 --hex-blob --set-gtid-purged=OFF  
--single-transaction --order-by-primary --flush-logs -q --databases db1 --tables  
t1 t2 >t1_t2.sql
```

3. 条件导出，导出 db1 表 t1 中 id=1 的数据。

```
mysqldump -uroot -p -P8635 -h192.168.0.199 --hex-blob --set-gtid-  
purged=OFF --single-transaction --order-by-primary --flush-logs -q --databases  
db1 --tables t1 --where='id=1'>t1_id.sql
```

4. 导出 db1 下所有表结构，而不导出数据。

```
mysqldump -uroot -p -P8635 -h192.168.0.199 --no-data --set-gtid-purged=OFF -  
--single-transaction --order-by-primary -n --flush-logs -q --databases  
db1 >db1_table.sql
```

5. 除 db1 下的表和数据外，其他对象全部导出。

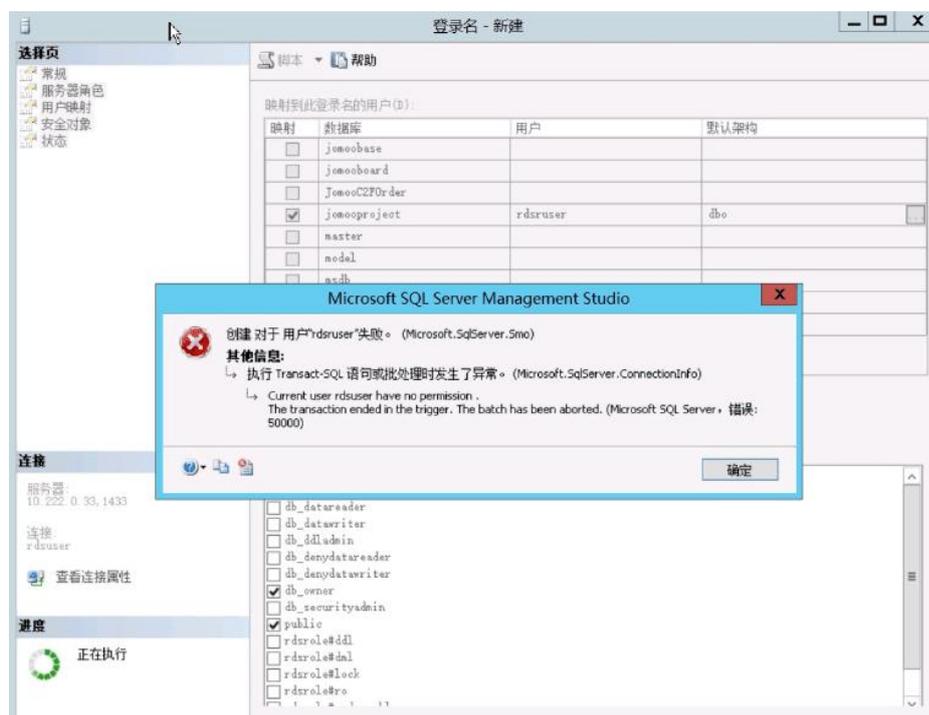
```
mysqldump -uroot -p -h192.168.0.199 -P8635 --set-gtid-purged=OFF -F -n -t -  
d -E -R db1 > others.sql
```

10

从阿里云迁移至天翼云的 Microsoft SQL Server 数据库无法创建用户

故障描述

从阿里云迁移至天翼云的 RDS Microsoft SQL Server 数据库，在创建用户时报错。



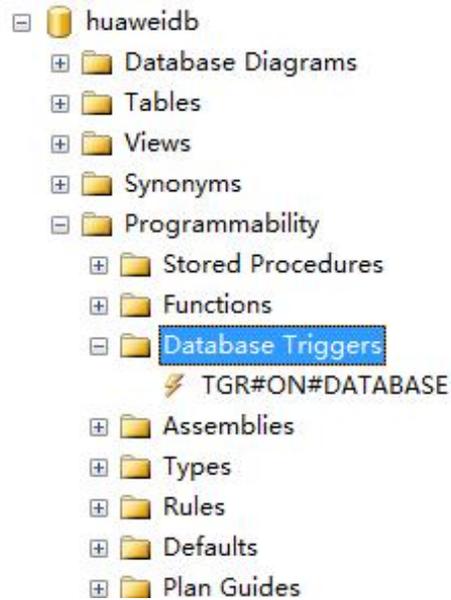
故障分析

阿里云 RDS Microsoft SQL Server 数据库，因存在阻止 SSMS 客户端创建用户并且限制授权的“TGR#ON#DATABASE”触发器，因而，只能通过阿里云管理界面创建数据库用户，不支持在 SSMS 客户端创建用户。天翼云支持在 SSMS 客户端创建数据库用户。

因而，从阿里云迁移到天翼云的 Microsoft SQL Server 数据库，需先禁用阿里云的“TGR#ON#DATABASE”触发器，再在 SSMS 客户端创建数据库用户并且授权。

步骤 1 登录 SSMS 客户端。

步骤 2 将“TGR#ON#DATABASE”触发器设置为“disable”，禁用“TGR#ON#DATABASE”触发器。



步骤 3 在 SSMS 客户端创建用户并授权。

⚠ 注意

新建的用户授权一定要映射 msdb 数据库。

---结束

11

MySQL 增加表字段后出现运行卡顿现象

故障描述

当给 MySQL 类型实例的表中增加一个字段，出现系统无法访问的现象。

解决方案

因增加表字段而引起数据库出现性能问题，有可能是未对新增字段添加索引，数据量大导致消耗了大量的 CPU 资源。为此，提出如下建议恢复数据库性能。

- 添加对应索引、主键。
- 优化慢 SQL 语句。

12

RDS 实例恢复到指定时间点失败

场景描述

使用恢复到指定时间点备份创建实例失败。

原因分析

备份时间戳出现偏差，导致恢复到指定时间点报错。

解决方案

进行全量备份恢复。

13

MySQL 数据库报错 Native error 1461 的解决方案

场景描述

MySQL 用户通常在并发读写、大批量插入 sql 语句或数据迁移等场景出现如下报错信息:

```
mysql_stmt_prepare failed! error(1461)Can't create more than  
max_prepared_stmt_count statements (current value: 16382)
```

故障分析

“max_prepared_stmt_count”的取值范围为 0~1048576，默认为“16382”，该参数限制了同一时间在 mysqld 上所有 session 中 prepared 语句的上限，用户业务超过了该参数当前值的范围。

解决方案

请您调大“max_prepared_stmt_count”参数的取值，建议调整为“65535”。

14 MySQL 创建表失败报错 Row size too large 的解决方案

场景描述

MySQL 用户创建表失败，出现如下报错信息：

Row size too large. The maximum row size for the used table type, not counting BLOBs, is 65535. This includes storage overhead, check the manual. You have to change some columns to TEXT or BLOBs

故障分析

“varchar” 的字段总和超过了 65535，导致创建表失败。

解决方案

1. 缩减长度，如下所示。

```
CREATE TABLE t1 (a VARCHAR(10000),b VARCHAR(10000),c VARCHAR(10000),d  
VARCHAR(10000),e VARCHAR(10000),f VARCHAR(10000) ) ENGINE=MyISAM CHARACTER SET  
latin1;
```

2. 请参考[官方文档](#)修改一个字段为 TEXT 类型。

15

MySQL 升级版本后，导致现有配置无法正常连接到 MySQL-server

场景描述

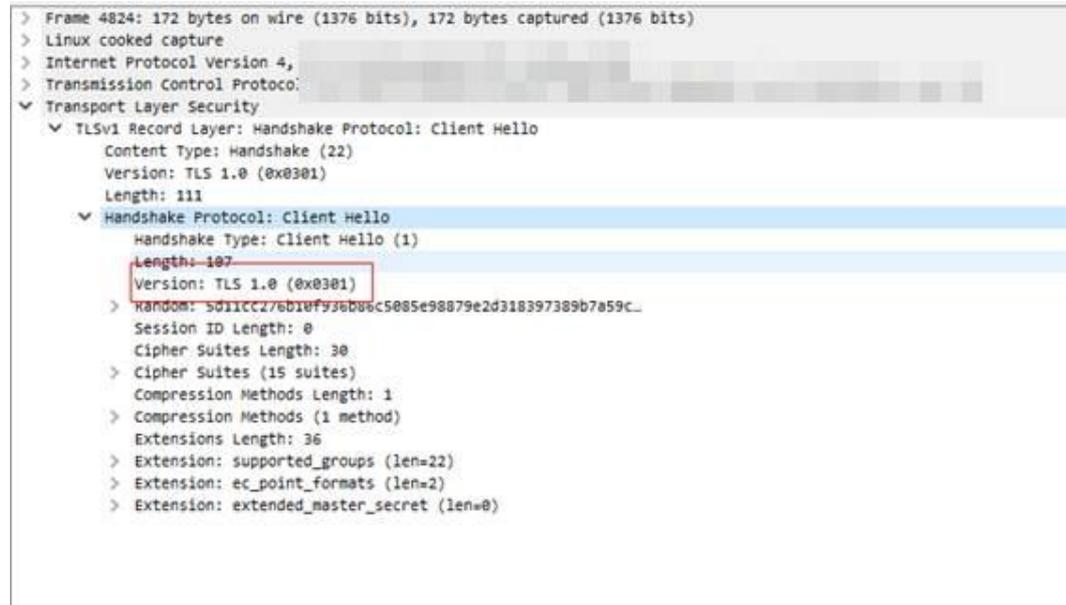
用户新建实例，用代码连接该数据库时出现报错：

Caused by: javax.net.ssl.SSLException: Received fatal alert: protocol_version

MySQL 原有版本为 5.7.23，升级到 5.7.25 版本后，导致现有配置无法正常连接到 MySQL-server，抓包结果如下图 15-1：

可以看出，客户端进行 TLS 握手时向服务端发送的 TLS 版本号是 1.0，并提供了 15 个支持的密码套件。

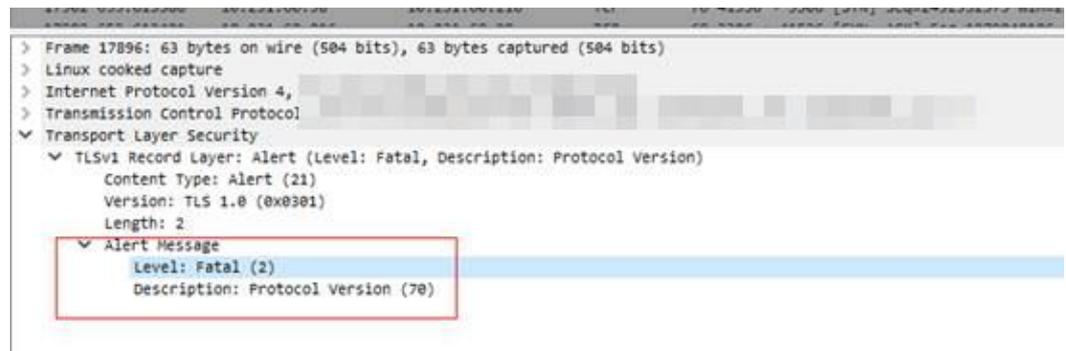
图 15-1 连接失败抓包结果



故障分析

从 MySQL-server 的回复中如图 15-2 可以看到，服务器拒绝了客户端的连接，原因是 MySQL 5.7.25 升级了 openssl 版本 (1.1.1a)，导致拒绝了不安全的 TLS 版本和密码套件。

图 15-2 MySQL-server 的回复



解决方案

升级您的 JDK 客户端到 **JDK 8 或以上版本**，则默认支持的 TLS 为 1.2 版本，如图 15-3，可以正常连接的客户端支持 TLS1.2，并支持 30 个密码套件。

图 15-3 正常连接抓包结果

65	1.406302	127.0.0.1	127.0.0.1	TLSv1.2	254 Client Hello
67	1.406235	127.0.0.1	127.0.0.1	TLSv1.2	2417 Server Hello, Certificate, Server Key Exchange, Certificate Request, Server Hello Done
69	1.407010	127.0.0.1	127.0.0.1	TLSv1.2	361 Certificate, Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
70	1.407284	127.0.0.1	127.0.0.1	TLSv1.2	298 New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
71	1.407385	127.0.0.1	127.0.0.1	TLSv1.2	274 Application Data
72	1.407454	127.0.0.1	127.0.0.1	TLSv1.2	96 Application Data
73	1.407549	127.0.0.1	127.0.0.1	TLSv1.2	322 Application Data
74	1.407691	127.0.0.1	127.0.0.1	TLSv1.2	379 Application Data


```

* Secure Sockets Layer
  * TLSv1.2 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 193
  * Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 189
    Version: TLS 1.2 (0x0303)
    Random: 3c5db7342ef3cf97facfa2b4dc0b5dcd9dc3435aece9a9f8...
    Session ID Length: 0
    Cipher Suites Length: 60
    Cipher Suites (50 suites)
  
```

16

磁盘空间满导致自动增量备份失败

场景描述

数据库未进行自动增量备份。

原因分析

随着业务数据的增加，实例负载太高，原来申请的数据库磁盘容量可能会不够用，尤其当实例显示“磁盘空间满”状态，且数据库不可进行写入操作，导致增量备份失败。

解决方案

您需要为天翼云关系型数据库实例进行扩容。在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 磁盘扩容”，进入“磁盘扩容”页面。磁盘扩容完成后进行一次全量备份，即可成功进行增量备份。

17

下载备份文件在本地恢复，登录密码错误

场景描述

下载备份文件在本地进行恢复，本地数据库登录密码提示错误，无法登录。

原因分析

下载备份文件在本地恢复，备份恢复之后，本地原自建数据库密码被云上数据库密码覆盖，导致使用本地原自建数据库密码无法登录。

解决方案

使用云上的 root 密码或在本地自建数据库重置密码。

A 修订记录

发布日期	修改说明
2020-04-10	第四次正式发布，包含以下内容： 新增 14 MySQL 创建表失败报错 Row size too large 的解决方案。
2019-11-30	第三次正式发布，包含以下内容： <ul style="list-style-type: none">• 新增 16 磁盘空间满导致自动增量备份失败。• 新增 17 下载备份文件在本地恢复，登录密码错误。
2019-09-30	第二次正式发布，包含以下内容： 新增 13 MySQL 数据库报错 Native error 1461 的解决方案。 新增 15 MySQL 升级版本后，导致现有配置无法正常连接到 MySQL-server。
2019-08-15	第一次正式发布。