

云数据库 RDS

用户指南

文档版本 01
发布日期 2025-10-28



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

1 产品介绍	1
1.1 什么是关系型数据库	1
1.2 常用概念	2
1.3 产品优势	3
1.3.1 低成本	3
1.3.2 高安全性	3
1.3.3 高可靠性	4
1.3.4 RDS 与自建数据库优势对比	5
1.4 产品系列	5
1.4.1 产品类型简介	5
1.4.2 产品功能对比	6
1.5 实例说明	7
1.5.1 数据库实例类型	7
1.5.2 数据库实例规格	8
1.5.3 数据库引擎和版本	10
1.5.4 数据库实例状态	10
1.6 典型应用	11
1.6.1 读写分离	11
1.7 约束与限制	11
1.7.1 MySQL 约束与限制	12
1.7.2 PostgreSQL 约束与限制	12
1.7.3 Microsoft SQL Server 约束与限制	13
1.8 RDS 与其他服务的关系	14
2 MySQL 快速入门	15
2.1 操作指引	15
2.2 步骤一：创建实例	16
2.3 步骤二：连接实例	20
2.3.1 实例连接方式介绍	20
2.3.2 通过内网连接 MySQL 实例	21
2.3.2.1 内网连接实例流程	21
2.3.2.2 设置安全组规则	22
2.3.2.3 通过内网连接 MySQL 实例	24
2.3.3 通过公网连接 MySQL 实例	27

2.3.3.1 公网连接实例流程.....	27
2.3.3.2 绑定弹性公网 IP.....	28
2.3.3.3 设置安全组规则.....	29
2.3.3.4 通过公网连接 MySQL 实例.....	31
3 PostgreSQL 快速入门.....	34
3.1 操作指引.....	34
3.2 步骤一：创建实例.....	35
3.3 步骤二：连接实例.....	38
3.3.1 实例连接方式介绍.....	38
3.3.2 通过内网连接 PostgreSQL 实例.....	40
3.3.2.1 内网连接实例流程.....	40
3.3.2.2 设置安全组规则.....	40
3.3.2.3 通过 psql 连接实例.....	42
3.3.3 通过公网连接 PostgreSQL 实例.....	44
3.3.3.1 公网连接实例流程.....	44
3.3.3.2 绑定弹性公网 IP.....	44
3.3.3.3 设置安全组规则.....	45
3.3.3.4 通过 psql 连接实例.....	46
4 SQL Server 快速入门.....	48
4.1 操作指引.....	48
4.2 步骤一：创建实例.....	49
4.3 步骤二：连接实例.....	52
4.3.1 实例连接方式介绍.....	52
4.3.2 通过内网连接 SQL Server 实例.....	54
4.3.2.1 内网连接实例流程.....	54
4.3.2.2 设置安全组规则.....	54
4.3.2.3 通过内网连接 SQL Server 实例.....	55
4.3.3 通过公网连接 SQL Server 实例.....	58
4.3.3.1 公网连接实例流程.....	58
4.3.3.2 绑定弹性公网 IP.....	59
4.3.3.3 设置安全组规则.....	60
4.3.3.4 通过公网连接 SQL Server 实例.....	61
5 MySQL 用户指南.....	64
5.1 数据迁移.....	64
5.1.1 使用 mysqldump 迁移 MySQL 数据.....	64
5.2 参数调优.....	67
5.2.1 MySQL 参数调优建议.....	67
5.3 MySQL 内核功能.....	69
5.3.1 内核版本说明.....	69
5.4 实例生命周期.....	75
5.4.1 创建与原实例相同配置的数据库实例.....	75

5.4.2 重启实例.....	76
5.4.3 自定义列表项.....	76
5.4.4 导出实例.....	77
5.4.5 删除实例.....	78
5.4.6 回收站.....	79
5.5 变更实例.....	80
5.5.1 大版本升级.....	80
5.5.2 修改实例名称.....	80
5.5.3 切换主备实例的可用性策略.....	81
5.5.4 变更实例的 CPU 和内存规格.....	81
5.5.5 磁盘扩容.....	82
5.5.6 设置可维护时间段.....	84
5.5.7 单机转主备实例.....	85
5.5.8 只读转单机实例.....	85
5.5.9 手动切换主备实例.....	86
5.5.10 迁移可用区.....	87
5.6 只读实例.....	88
5.6.1 只读实例简介.....	88
5.6.2 创建只读实例.....	89
5.6.3 管理只读实例.....	90
5.7 备份与恢复.....	91
5.7.1 备份概述.....	91
5.7.2 设置自动备份策略.....	92
5.7.3 设置跨区域备份策略.....	93
5.7.4 创建手动备份.....	94
5.7.5 下载备份文件.....	95
5.7.6 下载 Binlog 备份文件.....	96
5.7.7 设置 MySQL 本地 Binlog 日志清理.....	97
5.7.8 通过备份文件恢复关系型数据库实例数据.....	98
5.7.9 将数据库实例恢复到指定时间点.....	98
5.7.10 复制备份.....	99
5.8 参数模板管理.....	100
5.8.1 创建参数模板.....	100
5.8.2 编辑参数模板.....	101
5.8.3 导出参数模板.....	103
5.8.4 比较参数模板.....	104
5.8.5 查看参数修改历史.....	105
5.8.6 复制参数模板.....	106
5.8.7 重置参数模板.....	106
5.8.8 应用参数模板.....	107
5.8.9 查看参数模板应用记录.....	107
5.8.10 修改参数模板描述.....	108

5.8.11 删除参数模板.....	108
5.9 连接管理.....	109
5.9.1 配置和修改内网地址.....	109
5.9.2 绑定和解绑弹性公网 IP.....	110
5.9.3 修改数据库端口.....	111
5.9.4 下载证书.....	112
5.9.5 设置安全组规则.....	112
5.10 数据库管理.....	114
5.10.1 创建数据库.....	115
5.10.2 数据库授权.....	115
5.10.3 删除数据库.....	116
5.10.4 开启或关闭事件定时器.....	117
5.11 账号管理（非管理员权限）.....	118
5.11.1 创建数据库账号.....	118
5.11.2 重置数据库账号密码.....	119
5.11.3 修改数据库账号权限.....	119
5.11.4 删除数据库账号.....	120
5.12 数据库安全设置.....	120
5.13 数据安全性.....	121
5.13.1 重置管理员密码.....	121
5.13.2 修改实例安全组.....	122
5.14 监控指标与告警.....	123
5.14.1 支持的监控指标.....	123
5.14.2 设置告警规则.....	129
5.14.3 查看监控指标.....	129
5.15 CTS 审计.....	130
5.15.1 支持审计的关键操作列表.....	130
5.15.2 查看追踪事件.....	132
5.16 日志管理.....	132
5.16.1 查看或下载错误日志.....	132
5.16.2 查看或下载慢日志.....	133
5.16.3 查看主备切换日志.....	135
5.16.4 开启 SQL 审计日志.....	135
5.16.5 下载 SQL 审计日志.....	137
5.17 任务中心.....	138
5.17.1 查看任务.....	138
5.17.2 删除任务.....	139
6 PostgreSQL 用户指南.....	141
6.1 数据迁移.....	141
6.1.1 使用 psql 命令迁移 PostgreSQL 数据.....	141
6.2 参数调优.....	143
6.2.1 PostgreSQL 参数调优建议.....	143

6.3 PostgreSQL 增强版.....	144
6.3.1 PostgreSQL 增强版简介.....	144
6.3.2 函数.....	144
6.3.3 系统视图.....	149
6.3.4 数据类型.....	151
6.3.5 隐式类型转换.....	151
6.3.6 预定义参数.....	152
6.3.7 宏变量.....	152
6.3.8 操作符.....	153
6.3.9 语法.....	153
6.3.10 功能增强.....	155
6.3.11 安全加固.....	155
6.3.12 其他功能.....	156
6.4 PostgreSQL 内核功能.....	156
6.4.1 RDS for PostgreSQL 内核版本发布记录.....	156
6.5 实例生命周期.....	157
6.5.1 创建与原实例相同配置的数据库实例.....	157
6.5.2 重启实例.....	158
6.5.3 自定义列表项.....	158
6.5.4 导出实例.....	159
6.5.5 删除按需实例.....	159
6.5.6 回收站.....	161
6.6 变更实例.....	161
6.6.1 修改实例名称.....	162
6.6.2 切换主备实例的可用性策略.....	162
6.6.3 变更实例的 CPU 和内存规格.....	163
6.6.4 磁盘扩容.....	164
6.6.5 单机转主备实例.....	165
6.6.6 手动切换主备实例.....	166
6.6.7 迁移可用区.....	167
6.7 只读实例.....	168
6.7.1 只读实例简介.....	168
6.7.2 创建只读实例.....	169
6.7.3 管理只读实例.....	171
6.8 备份与恢复.....	171
6.8.1 备份概述.....	171
6.8.2 设置自动备份策略.....	172
6.8.3 设置跨区域备份策略.....	173
6.8.4 创建手动备份.....	174
6.8.5 下载全量备份文件.....	175
6.8.6 下载增量备份文件.....	176
6.8.7 通过备份文件恢复关系型数据库实例数据.....	176

6.8.8 将数据库实例恢复到指定时间点.....	177
6.8.9 复制备份.....	179
6.9 参数模板管理.....	179
6.9.1 创建参数模板.....	179
6.9.2 修改实例参数.....	180
6.9.3 导出参数模板.....	182
6.9.4 比较参数模板.....	183
6.9.5 查看参数修改历史.....	184
6.9.6 复制参数模板.....	185
6.9.7 重置参数模板.....	185
6.9.8 应用参数模板.....	186
6.9.9 查看参数模板应用记录.....	186
6.9.10 修改参数模板描述.....	187
6.9.11 删除参数模板.....	187
6.10 连接管理.....	188
6.10.1 配置和修改内网地址.....	188
6.10.2 绑定和解绑弹性公网 IP.....	189
6.10.3 修改数据库端口.....	190
6.10.4 通过 pgAdmin 连接实例.....	190
6.11 插件管理.....	193
6.11.1 创建和删除插件.....	193
6.11.2 支持的插件列表.....	195
6.12 数据库安全设置.....	198
6.13 数据安全性.....	199
6.13.1 重置管理员密码.....	199
6.13.2 修改实例安全组.....	200
6.14 与告警监控指标与告警.....	201
6.14.1 支持的监控指标.....	201
6.14.2 设置告警规则.....	204
6.14.3 查看监控指标.....	205
6.15 CTS 审计.....	206
6.15.1 支持审计的关键操作列表.....	206
6.15.2 查看追踪事件.....	207
6.16 任务中心.....	208
6.16.1 查看任务.....	208
6.16.2 删除任务.....	209
7 SQL Server 用户指南.....	210
7.1 实例生命周期.....	210
7.1.1 创建与原实例相同配置的数据库实例.....	210
7.1.2 重启实例.....	211
7.1.3 自定义列表项.....	211
7.1.4 导出实例.....	212

7.1.5 删除按需实例.....	212
7.1.6 回收站.....	214
7.2 变更实例.....	214
7.2.1 修改实例名称.....	215
7.2.2 切换主备实例的可用性策略.....	215
7.2.3 变更实例的 CPU 和内存规格.....	216
7.2.4 磁盘扩容.....	217
7.2.5 单机转主备实例.....	218
7.2.6 手动切换主备实例.....	218
7.3 只读实例.....	219
7.3.1 只读实例简介.....	219
7.3.2 创建只读实例.....	220
7.3.3 管理只读实例.....	222
7.4 备份与恢复.....	223
7.4.1 备份概述.....	223
7.4.2 设置自动备份策略.....	223
7.4.3 创建手动备份.....	224
7.4.4 下载备份文件.....	225
7.4.5 通过备份文件恢复关系型数据库实例数据.....	226
7.4.6 复制备份.....	228
7.5 参数模板管理.....	228
7.5.1 创建参数模板.....	228
7.5.2 编辑参数模板.....	229
7.5.3 导出参数模板.....	231
7.5.4 比较参数模板.....	232
7.5.5 查看参数修改历史.....	233
7.5.6 复制参数模板.....	234
7.5.7 重置参数模板.....	234
7.5.8 应用参数模板.....	235
7.5.9 查看参数模板应用记录.....	235
7.5.10 修改参数模板描述.....	236
7.5.11 删除参数模板.....	236
7.6 连接管理.....	237
7.6.1 配置和修改内网地址.....	237
7.6.2 绑定和解绑弹性公网 IP.....	238
7.6.3 修改数据库端口.....	239
7.7 数据安全性.....	240
7.7.1 重置管理员密码.....	240
7.7.2 修改实例安全组.....	241
7.8 与告警监控指标与告警.....	241
7.8.1 支持的监控指标.....	242
7.8.2 设置告警规则.....	246

7.8.3 查看监控指标.....	247
7.9 CTS 审计.....	247
7.9.1 支持审计的关键操作列表.....	247
7.9.2 查看追踪事件.....	249
7.10 任务中心.....	250
7.10.1 查看任务.....	250
7.10.2 删除任务.....	251
7.11 存储过程的使用.....	252
7.11.1 创建主账号.....	253
7.11.2 更新警报和作业所用的操作员信息.....	253
7.11.3 删除警报.....	255
7.11.4 删除特定警报和操作员的 SQL Server 代理通知定义.....	256
7.11.5 删除操作员.....	257
8 常见问题.....	259
8.1 产品咨询.....	259
8.1.1 使用 RDS 要注意些什么.....	259
8.1.2 什么是 RDS 实例可用性.....	259
8.1.3 可以通过创建模板的方式创建实例吗.....	259
8.1.4 RDS 与其他数据库解决方案间的差异.....	260
8.1.5 RDS 实例是否会受其他用户实例的影响.....	260
8.1.6 关系型数据库支持跨 AZ 高可用吗.....	260
8.1.7 RDS 是否支持主备实例变更为单机实例.....	260
8.1.8 导出 SQL 查询结果到 Excel 出现乱码.....	260
8.1.9 如何搭建 AD 域.....	261
8.1.10 为何使用了 RDS 后网站登录较慢.....	266
8.1.11 云数据库如何进行主备切换.....	266
8.1.12 多台弹性云服务器是否可以使用同一个 RDS 数据库.....	267
8.1.13 删除 RDS SQL Server 主备实例数据库报错.....	267
8.2 资源及磁盘管理.....	268
8.2.1 占用 RDS 磁盘空间的日志及文件有哪些.....	268
8.2.2 哪些内容会占用用户所创建的 RDS 实例空间.....	269
8.2.3 用户申请的 RDS 实例磁盘空间会有哪些系统开销.....	269
8.2.4 用户必须进行 DDL 操作时对磁盘空间的要求.....	269
8.2.5 使用 RDS 可以运行多少个 RDS 实例.....	269
8.2.6 在一个 RDS 实例内能运行多少个数据库.....	270
8.3 数据库连接.....	270
8.3.1 外部服务器能否访问 RDS 数据库.....	270
8.3.2 RDS 数据库连接数满的排查思路.....	270
8.3.3 RDS 数据库实例支持的最大数据连接数是多少.....	270
8.3.4 如何创建和连接 ECS.....	272
8.3.5 内网方式下 ECS 无法连接 RDS 实例的原因.....	272
8.3.6 客户端问题导致连接失败.....	273

8.3.7 服务端问题导致连接失败.....	273
8.3.8 应用程序怎么访问 VPC 中的 RDS 实例.....	273
8.3.9 应用程序是否需要支持自动重连数据库.....	274
8.3.10 如何通过 JDBC 连接 PostgreSQL 数据库.....	274
8.3.11 SQL Server 连接不上的判断方法.....	277
8.3.12 RDS 跨地域内网能访问吗.....	277
8.3.13 使用 SSL 连接数据库发生主备切换后是否可以正常连接.....	277
8.3.14 MySQL 支持 SSL 连接吗.....	277
8.3.15 为什么重置密码后新密码没有生效.....	278
8.4 数据库迁移.....	278
8.4.1 为什么要采用 mysqldump 和 pg_dump 工具迁移.....	278
8.4.2 RDS 支持导入哪些数据库引擎的数据.....	278
8.5 数据库权限.....	278
8.5.1 root 账号为什么没有 super 权限.....	278
8.6 数据库存储.....	279
8.6.1 RDS for MySQL 支持哪些存储引擎.....	279
8.6.2 RDS 存储的存储配置是什么.....	280
8.6.3 RDS 是否支持存储类型变更, 普通 IO 转超高 IO.....	280
8.6.4 数据超过了 RDS 实例的最大存储容量怎么办.....	280
8.7 安装客户端.....	281
8.7.1 如何安装 MySQL 客户端.....	281
8.7.2 如何安装 PostgreSQL 客户端.....	281
8.7.3 如何安装 SQL Server Management Studio.....	283
8.8 备份与恢复.....	284
8.8.1 RDS 能够保存多长时间的备份.....	284
8.8.2 数据库在备份窗口中是否可用.....	284
8.8.3 如何将 RDS 数据库备份到上.....	284
8.8.4 为什么自动备份会失败.....	284
8.8.5 删除 RDS 实例后数据库备份会怎样.....	285
8.8.6 如果删除云账号备份是否会被删除.....	285
8.8.7 为什么数据表会丢失或数据被删除.....	285
8.9 数据库监控.....	285
8.9.1 用户平时需要关注实例的哪些监控指标.....	285
8.10 扩容及规格变更.....	286
8.10.1 RDS 实例在扩容期间是否一直可用.....	286
8.10.2 为什么端口变更后实例状态异常.....	286
8.11 数据库参数修改.....	286
8.11.1 哪些参数设置不合理会导致数据库不可用 (PostgreSQL)	286
8.11.2 如何设置 SQL Server 添加 ndf 文件的路径.....	287
8.11.3 RDS for MySQL 密码过期策略.....	287
8.12 日志管理.....	288
8.12.1 RDS for MySQL 慢日志延迟时间是多久.....	288

8.12.2 RDS for SQL Server 慢查询需要多长时间.....	288
8.12.3 如何通过命令获取 Microsoft SQL Server 的错误日志.....	289
8.12.4 RDS 慢日志统计分析是否可以导出.....	289
8.13 网络安全.....	289
8.13.1 RDS 有哪些安全保障措施.....	289
8.13.2 如何确保在 VPC 中运行的 RDS 实例的安全.....	290
8.13.3 如何保障 EIP 数据传输安全.....	290
8.13.4 如何防止任意源连接数据库.....	290
8.13.5 将根证书导入 Windows/Linux 操作系统.....	290
8.13.6 如何确认 SSL 根证书的有效期限.....	291
8.13.7 如何甄别数据库中数据被损坏.....	291
9 故障排除.....	292
9.1 RDS for MySQL.....	292
9.1.1 备份恢复.....	292
9.1.1.1 备份期间 DDL 操作导致备份失败.....	292
9.1.1.2 RDS for MySQL 备份任务失败分析思路.....	292
9.1.1.3 手动下发全量备份比自动下发全量备份时间长.....	294
9.1.1.4 磁盘空间满导致自动增量备份失败.....	294
9.1.1.5 使用 mysqldump 导出大表的注意事项.....	295
9.1.1.6 mysqldump 的 6 大使用场景的导出命令.....	295
9.1.1.7 怎么解决执行 mysqldump 出现 SET @@SESSION.SQL_LOG_BIN 等 SQL 的问题.....	296
9.1.1.8 mysqldump 导出数据报错权限不足.....	297
9.1.2 主备复制.....	298
9.1.2.1 MySQL 主备复制原理简介.....	298
9.1.2.2 主备复制延迟持续增长后自动恢复.....	299
9.1.2.3 MySQL 主备复制延迟场景及解决方案.....	300
9.1.2.4 RDS 主备复制关系异常.....	302
9.1.2.5 主备复制时延瞬间飆高回落.....	302
9.1.3 参数类.....	303
9.1.3.1 控制台上修改 long_query_time 参数后未生效.....	303
9.1.3.2 GROUP_CONCAT 结果不符合预期.....	303
9.1.3.3 RDS for MySQL 创建索引失败报错[ERROR] 1071 的解决方案.....	304
9.1.3.4 RDS for MySQL 大小写参数敏感类问题.....	306
9.1.3.5 RDS MySQL timeout 相关参数简介.....	307
9.1.4 性能资源类.....	308
9.1.4.1 数据库磁盘满导致被设置 read_only.....	308
9.1.4.2 Binlog 未清理导致磁盘占用高.....	308
9.1.4.3 MySQL 只读实例磁盘占用远超主实例.....	309
9.1.4.4 冷热数据问题导致 sql 执行速度慢.....	310
9.1.4.5 表空间膨胀问题.....	310
9.1.4.6 复杂查询造成磁盘满.....	311
9.1.4.7 长事务导致规格变更或小版本升级失败.....	312

9.1.4.8 RDS for MySQL 数据库报错 Native error 1461 的解决方案.....	313
9.1.4.9 RDS for MySQL 增加表字段后出现运行卡顿现象.....	313
9.1.4.10 长事务导致 UNDO 增多引起磁盘空间满.....	313
9.1.5 SQL 类.....	314
9.1.5.1 更新 emoji 表情数据报错 Error 1366.....	314
9.1.5.2 建表时 timestamp 字段默认值无效.....	315
9.1.5.3 自增属性 AUTO_INCREMENT 为什么未在表结构中显示.....	315
9.1.5.4 存储过程和相关表字符集不一致导致执行缓慢.....	316
9.1.5.5 RDS MySQL 报错 ERROR [1412]的解决方法.....	317
9.1.5.6 创建二级索引报错 Too many keys specified.....	317
9.1.5.7 存在外键的表删除问题.....	318
9.1.5.8 distinct 与 group by 优化.....	319
9.1.5.9 字符集和字符序的默认选择方式.....	320
9.1.5.10 MySQL 创建用户提示服务器错误.....	322
9.1.5.11 delete 大表数据后，再次查询同一张表时出现慢 SQL.....	323
9.1.5.12 为什么有时候用浮点数做等值比较查不到数据.....	324
9.1.5.13 有外键的表无法删除报错 ERROR[1451]的解决方案.....	326
9.1.5.14 表字段类型转换失败的解决方法.....	327
9.1.5.15 RDS for MySQL 创建表失败报错 Row size too large 的解决方案.....	327
9.1.5.16 RDS for MySQL 数据库报错 ERROR [1412]的解决方案.....	328
9.1.6 连接类.....	328
9.1.6.1 连接数据库报错 Access denied.....	328
9.1.6.2 RDS for MySQL 建立连接慢导致客户端超时报 connection established slowly.....	330
9.1.6.3 客户端 TLS 版本与 RDS for MySQL 不一致导致 SSL 连接失败.....	331
9.1.6.4 RDS for MySQL 客户端连接实例后会自动断开.....	332
9.1.6.5 RDS for MySQL 实例无法访问.....	333
9.1.6.6 RDS for MySQL 数据库修改 authentication_string 字段为显示密码后无法登录.....	336
9.1.6.7 RDS for MySQL 升级版本后，导致现有配置无法正常连接到 MySQL-server.....	337
9.1.6.8 客户端超时参数设置不当导致连接超时退出.....	338
9.1.6.9 RDS for MySQL 在启用了 SSL 验证连接功能后，导致代码（php/java/python）等连接数据库失败.....	339
9.1.7 其他使用问题.....	339
9.1.7.1 自增字段取值.....	339
9.1.7.2 表的自增 AUTO_INCREMENT 初值与步长.....	341
9.1.7.3 自增字段值跳变的原因.....	344
9.1.7.4 修改表的自增 AUTO_INCREMENT 值.....	349
9.1.7.5 自增主键达到上限，无法插入数据.....	351
9.1.7.6 空用户的危害.....	353
9.1.7.7 pt-osc 工具连接 RDS for MySQL 主备实例卡住.....	355
9.2 RDS for SQL Server.....	356
9.2.1 RDS for SQL Server 数据库恢复到新实例后新建用户，提示用户已存在的解决方法.....	357

1 产品介绍

1.1 什么是关系型数据库

关系型数据库（Relational Database Service，简称RDS）是一种基于云计算平台的稳定可靠、弹性伸缩、便捷管理的在线关系型数据库服务。关系型数据库支持以下引擎：

- [MySQL](#)
- [PostgreSQL](#)
- [SQL Server](#)

关系型数据库服务具有完善的性能监控体系和多重安全防护措施，并提供了专业的数据库管理平台，让用户能够在云上轻松地进行设置和扩展关系型数据库。通过关系型数据库服务的管理控制台，用户无需编程就可以执行所有必需任务，简化运营流程，减少日常运维工作量，从而专注于开发应用和业务发展。

MySQL

MySQL是目前最受欢迎的开源数据库之一，其性能卓越，搭配LAMP（Linux + Apache + MySQL + Perl/PHP/Python），成为WEB开发的高效解决方案。云数据库MySQL拥有稳定可靠、安全运行、弹性伸缩、轻松管理、经济实用等特点。

- 架构成熟稳定，支持流行应用程序，适用于多领域多行业；支持各种WEB应用，成本低，中小企业首选。
- 管理控制台提供全面的监控信息，简单易用，灵活管理，可视又可控。
- 随时根据业务情况弹性伸缩所需资源，按需开支，量身订做。

当前RDS for MySQL支持版本请参见[数据库引擎和版本](#)。

PostgreSQL

PostgreSQL是一个开源对象关系型数据库管理系统，并侧重于可扩展性和标准的符合性，被业界誉为“最先进的开源数据库”。云数据库PostgreSQL面向企业复杂SQL处理的OLTP在线事务处理场景，支持NoSQL数据类型（JSON/XML/hstore），支持GIS地理信息处理，在可靠性、数据完整性方面有良好声誉，适用于互联网网站、位置应用系统、复杂数据对象处理等应用场景。

- 支持postgis插件，空间应用卓越。
- 适用场景丰富，费用低，随时可以根据业务情况弹性伸缩所需的资源，按需开支，量身订做。

当前RDS for PostgreSQL支持版本请参见[数据库引擎和版本](#)。

Microsoft SQL Server

SQL Server是商用级数据库，成熟的企业级架构，轻松应对各种复杂环境。一站式部署、保障关键运维服务，大量降低人力成本。被广泛应用于政府、金融、医疗、教育和游戏等领域。

云数据库SQL Server具有稳定可靠、安全运行、弹性伸缩、轻松管理和经济实用等特点。拥有高可用架构、数据安全保障和故障秒级恢复功能，提供了灵活的备份方案。

当前RDS for SQL Server支持版本请参见[数据库引擎和版本](#)。

1.2 常用概念

实例

关系型数据库的最小管理单元是实例，一个实例代表了一个独立运行的数据库。用户可以在关系型数据库系统中自助创建及管理各种数据库引擎的实例。实例的类型、规格、引擎、版本和状态，请参考[实例说明](#)。

数据库引擎

关系型数据库支持以下引擎：

- MySQL
- PostgreSQL
- SQL Server

支持版本请参见[数据库引擎和版本](#)。

实例类型

各系列产品的详细介绍请参考[产品类型简介](#)和[产品功能对比](#)。

实例规格

数据库实例各种规格（vCPU个数、内存（GB）、对应的数据库引擎）请参考[数据库实例规格](#)。

自动备份

创建实例时，关系型数据库默认开启自动备份策略，实例创建成功后，您可对其进行修改，关系型数据库会根据您的配置，自动创建数据库实例的全量备份。

手动备份

手动备份是由用户启动的数据库实例的全量备份，它会一直保存，直到用户手动删除。

区域和可用区

我们用区域和可用区来描述数据中心的位置，您可以在特定的区域、可用区创建资源。

- 区域（Region）指物理的数据中心。每个区域完全独立，这样可以实现最大程度的容错能力和稳定性。资源创建成功后不能更换区域。
- 可用区（AZ, Availability Zone）是同一区域内，电力和网络互相隔离的物理区域，一个可用区不受其他可用区故障的影响。一个区域内可以有多个可用区，不同可用区之间物理隔离，但内网互通，既保障了可用区的独立性，又提供了低价、低时延的网络连接。

阐明了区域和可用区之间的关系。

项目

Project用于将OpenStack的资源（计算、存储和网络资源）进行分组和隔离。Project可以是一个部门或者一个项目组。一个账户中可以创建多个Project。

1.3 产品优势

1.3.1 低成本

创建使用

您可以通过官网实时生成目标实例，关系型数据库服务配合弹性云服务器一起使用，通过内网连接关系型数据库可以有效地降低应用响应时间、节省公网流量费用。

弹性扩容

可以根据您的业务情况弹性伸缩所需的资源，按需开支，量身定做。配合云监控（Cloud Eye）监测数据库压力和数据存储量的变化，您可以灵活调整实例规格。

完全兼容

您无需再次学习，关系型数据库各引擎的操作方法与原生数据库引擎的完全相同。关系型数据库还兼容现有的程序和工具。

运维便捷

RDS的日常维护和管理，包括但不限于软硬件故障处理、数据库补丁更新等工作，保障关系型数据库运转正常。云数据库RDS提供专业数据库管理平台，重启、重置密码、参数修改、查看错误日志和慢查询日志、恢复数据等一键式功能。提供CPU利用率、IOPS、连接数、磁盘空间等实例信息实时监控及告警，让您随时随地了解实例动态。

1.3.2 高安全性

网络隔离

通过虚拟私有云（Virtual Private Cloud，简称VPC）和网络安全组实现网络隔离。虚拟私有云允许租户通过配置虚拟私有云入站IP范围，来控制连接数据库的IP地址段。关

关系型数据库实例运行在租户独立的虚拟私有云内，可提升关系型数据库实例的安全性。您可以综合运用子网和安全组的配置，来完成关系型数据库实例的隔离。

访问控制

通过主/子账号和安全组实现访问控制。创建关系型数据库实例时，关系型数据库服务会为租户同步创建一个数据库主账户，根据需要创建数据库实例和数据库子账户，将数据库对象赋予数据库子账户，从而达到权限分离的目的。可以通过虚拟私有云对关系型数据库实例所在的安全组入站、出站规则进行限制，从而控制可以连接数据库的网络范围。

传输加密

通过TLS加密、SSL加密实现传输加密。使用从服务控制台下载CA根证书，并在连接数据库时提供该证书，对数据库服务端进行认证并达到加密传输的目的。

存储加密

通过静态加密、表空间加密对数据进行加密。关系型数据库服务支持对存储到数据库中的数据加密后存储，加密密钥由进行管理。

数据删除

删除关系型数据库实例时，存储在数据库实例中的数据都会被删除。安全删除不仅包括数据库实例所挂载的磁盘，也包括自动备份数据的存储空间。删除的实例可以通过保留的手动备份恢复实例数据。

防 DDoS 攻击

当用户使用外网连接关系型数据库实例时，可能会遭受DDoS攻击。当关系型数据库安全体系认为用户实例正在遭受DDoS攻击时，会首先启动流量清洗的功能，如果流量清洗无法抵御攻击或者攻击达到黑洞阈值时，将会进行黑洞处理，保证关系型数据库整体服务的可用性。

安全防护

关系型数据库处于多层防火墙的保护之下，可以有力地抗击各种恶意攻击，保证数据安全，防御DDoS攻击、防SQL注入等。建议用户通过内网访问关系型数据库实例，可使关系型数据库实例免受DDoS攻击风险。

1.3.3 高可靠性

双机热备

关系型数据库服务采用热备架构，故障秒级自动切换。

数据备份

每天自动备份数据，上传到对象存储服务（Object Storage Service，简称OBS）。备份文件保留732天，支持一键式恢复。用户可以设置自动备份的周期，还可以根据自身业务特点随时发起备份，选择备份周期、修改备份策略。

数据恢复

支持按备份集和指定时间点的恢复。在大多数场景下，用户可以将732天内任意一个时间点的数据恢复到关系型数据库新实例上，数据验证无误后即可将数据迁回关系型数据库主实例，完成数据回溯。

1.3.4 RDS 与自建数据库优势对比

性能对比

性能项目	云数据库RDS	自购服务器搭建数据库服务
服务可用性	请参见《弹性云服务器用户指南》。	需要购买额外设备，自建主从，自建RAID。
数据可靠性	请参见《云硬盘用户指南》。	需要购买额外设备，自建主从，自建RAID。
系统安全性	防DDoS攻击，流量清洗；及时修复各种数据库安全漏洞。	需要购买昂贵的硬件设备和软件服务，需要自行检测和修复安全漏洞等。
数据库备份	支持自动备份，手动备份，自定义备份存储周期。	需要购买设备，并自行搭建设置和后期维护。
软硬件投入	无需投入软硬件成本，按需购买，弹性伸缩。	数据库服务器成本相对较高，对于SQL Server需支付许可证费用。
系统托管	无需托管。	需要自购2U服务器设备，如需实现主从，购买两台服务器，并进行自建。
维护成本	无需运维。	需要投入大量人力成本，招聘专业的DBA进行维护。
部署扩容	弹性扩容，快速升级，按需开通。	需采购和原设备匹配的硬件，需托管机房的配合，需部署设备，整体周期较长。
资源利用率	按实际结算，100%利用率。	考虑峰值，资源利用率低。

1.4 产品系列

1.4.1 产品类型简介

目前，云数据库RDS的实例分为如下几个类型：

- 单机实例
- 主备实例
- 集群版实例

不同系列支持的引擎类型和实例规格不同，请以实际界面为准。

表 1-1 实例类型简介

实例类型	简介	适用场景
单机实例	采用单个数据库节点部署架构。与主流的主备实例相比，它只包含一个节点，但具有高性价比。	<ul style="list-style-type: none">个人学习。微型网站。中小企业的开发测试环境。
主备实例	采用一主一备的经典高可用架构，支持跨AZ高可用，可以选择主可用区和备可用区不在同一个可用区（AZ）。 主实例和备实例共用一个IP地址。	<ul style="list-style-type: none">大中型企业的生产数据库。覆盖互联网、物联网、零售电商、物流、游戏等行业的应用。
集群版实例	采用微软AlwaysOn高可用架构，支持1主1备5只读集群模式，拥有更高可用性，可靠性，可拓展能力。	<ul style="list-style-type: none">金融行业。互联网行业。酒店行业。在线教育。

1.4.2 产品功能对比

单机实例采用单个数据库节点部署架构。与主流的主备实例相比，单机实例只包含一个节点，没有备用节点用于故障恢复。

优势对比

- 单机实例：支持创建只读实例、错误日志、慢日志查询管理。相较于主备实例，单机实例少了一个数据库节点，可大幅节省用户成本。由于单机实例只有一个数据库节点，当该数据库节点出现故障时，恢复时间较长，因此，如果是对数据库可用性要求较高的敏感性业务，不建议使用单机实例。
- 主备实例：主备实例的备数据库节点仅用于故障转移和恢复场景，不对外提供服务。由于使用备数据库节点会带来额外性能开销，从性能角度来看，单机实例的性能与主备实例相同，甚至单机实例的性能可能会高于主备实例。
- 集群版实例：采用微软AlwaysOn高可用架构，支持1主1备5只读集群模式，拥有更高可用性，可靠性，可拓展能力。

表 1-2 产品功能对比

产品功能	单机实例	主备实例	集群版实例
节点数	1	2	2

产品功能	单机实例	主备实例	集群版实例
规格配置	vCPU：最高60核 内存大小：最高512GB 数据盘：最高4000GB	vCPU：最高60核 内存大小：最高512GB 数据盘：最高4000GB	vCPU：最高64核 内存大小：最高512GB 数据盘：最高4TB
监控与告警	支持	支持	支持
安全组	支持	支持	支持
备份与恢复	支持	支持	支持
回收站	支持	支持	支持
参数设置	支持	支持	支持
SSL	支持	支持	支持
日志管理	支持	支持	支持
只读实例（需另建实例）	支持	支持	支持
高频监控	支持	支持	支持
主备库切换	不支持	支持	支持
可用区切换	不支持	支持	支持
手动主备切换	不支持	支持	支持
实例规格变更	支持	支持	支持

1.5 实例说明

1.5.1 数据库实例类型

数据库实例是关系型数据库的最小管理单元。一个实例代表了一个独立运行的关系型数据库。您可以在一个实例中创建和管理多个数据库，并且可以使用与独立访问数据库实例相同的工具和应用进行访问。使用管理控制台或API可以方便地创建或者修改数据库实例。关系型数据库服务对运行实例数量没有限制，但每个数据库实例都有唯一的标识符。

实例可进行如下分类：

表 1-3 实例类型

实例类型	简介
单机实例	采用单个数据库节点部署架构。与主流的主备实例相比，它只包含一个节点，但具有高性价比。
主备实例	采用一主一备的经典高可用架构，主备实例的每个节点的规格保持一致。
只读实例	采用单个物理节点的架构（没有备节点）。
集群版实例	采用微软AlwaysOn高可用架构，支持1主1备5只读集群模式，拥有更高可用性，可靠性，可拓展能力。

用户可以在关系型数据库系统中自助创建及管理各种数据库引擎的实例。

不同实例类型之间的区别和功能对比请参考[产品类型简介](#)和[产品功能对比](#)。

1.5.2 数据库实例规格

通用增强II型实例性能强劲稳定，搭载全新网络加速引擎，以及DPDK(Data Plane Development Kit)快速报文处理机制，提供更高的网络性能，满足不同场景需求。

通用增强II型实例搭载第二代英特尔® 至强® 可扩展处理器，多项技术优化，计算性能强劲稳定，配套25GE智能高速网卡，提供超高网络带宽和PPS收发包能力。

表 1-4 性能规格

规格	vCPU(个)	内存(GB)	支持的数据库引擎
通用增强II型	1	2	<ul style="list-style-type: none">MySQLPostgreSQL
	1	4	<ul style="list-style-type: none">MySQLPostgreSQL
	1	8	MySQL
	2	4	MySQL
	2	8	<ul style="list-style-type: none">MySQLPostgreSQLMicrosoft SQL Server
	2	16	<ul style="list-style-type: none">MySQLPostgreSQLMicrosoft SQL Server

规格	vCPU(个)	内存(GB)	支持的数据库引擎
	4	8	<ul style="list-style-type: none"> MySQL PostgreSQL Microsoft SQL Server
	4	16	<ul style="list-style-type: none"> MySQL Microsoft SQL Server
	4	32	<ul style="list-style-type: none"> MySQL Microsoft SQL Server
	8	32	<ul style="list-style-type: none"> MySQL PostgreSQL Microsoft SQL Server
	8	64	<ul style="list-style-type: none"> MySQL Microsoft SQL Server
	16	32	MySQL
	16	64	<ul style="list-style-type: none"> MySQL PostgreSQL Microsoft SQL Server
	16	128	<ul style="list-style-type: none"> MySQL Microsoft SQL Server
	24	192	Microsoft SQL Server
	32	64	MySQL
	32	128	<ul style="list-style-type: none"> MySQL PostgreSQL Microsoft SQL Server
	60	128	<ul style="list-style-type: none"> MySQL PostgreSQL
	60	256	<ul style="list-style-type: none"> MySQL PostgreSQL Microsoft SQL Server
	60	512	MySQL

数据库实例规格请以实际环境为准。

1.5.3 数据库引擎和版本

关系型数据库服务目前支持的数据库引擎和版本如表1-5所示。

新应用上线，建议您使用数据库引擎对应的最新大版本，以MySQL为例，建议您选择MySQL 5.7。用户创建实例时，不可选择小版本，如MySQL 5.7.31，RDS会提供最优的小版本，实例创建成功，您可在console“实例管理”页面实例列表中的“数据库引擎版本”列，查看具体的小版本号。数据库引擎和版本请以实际环境为准。

表 1-5 数据库引擎和版本

数据库引擎	单机实例	主备实例	集群版实例
MySQL引擎	<ul style="list-style-type: none">● 8.0● 5.7● 5.6	<ul style="list-style-type: none">● 8.0● 5.7● 5.6	暂不支持
PostgreSQL引擎	<ul style="list-style-type: none">● 13● 增强版，即1.0版本，更多介绍请参见PostgreSQL增强版。● 12● 11● 10● 9.6● 9.5	<ul style="list-style-type: none">● 13● 增强版，即1.0版本，更多介绍请参见PostgreSQL增强版。● 12● 11● 10● 9.6● 9.5	暂不支持
Microsoft SQL Server引擎	<ul style="list-style-type: none">● 2019 企业版● 2019 标准版● 2017 企业版● 2017 标准版● 2016 企业版● 2016 标准版● 2014 标准版● 2014 企业版	<ul style="list-style-type: none">● 2019 标准版● 2017 企业版● 2017 标准版● 2016 企业版● 2016 标准版● 2014 企业版● 2014 标准版	<ul style="list-style-type: none">● 2019 企业版● 2017 企业版

1.5.4 数据库实例状态

数据库实例状态

数据库实例状态是数据库实例的运行情况。用户可以使用管理控制台和API操作查看数据库实例状态。

表 1-6 状态及说明

状态	说明
正常	数据库实例正常和可用。
异常	数据库实例不可用。
创建中	正在创建数据库实例。
创建失败	数据库实例创建失败。
主备切换中	正在进行主实例和备实例的切换。
转主备中	单机实例正在转换为主备实例。
重启中	实例重启中。
端口修改中	正在修改数据库实例的数据库端口。
规格变更中	数据库实例的CPU和内存规格变更中。
扩容中	数据库实例的磁盘空间扩容中。
备份中	正在备份数据库实例。
恢复中	正在恢复备份到实例中。
恢复失败	实例恢复失败。
存储空间满	实例的磁盘空间已满，此时不可进行数据库写入操作，您需要扩容磁盘使实例恢复到正常状态。
已删除	数据库实例已被删除，对于已经删除的实例，将不会在实例列表中显示。
等待重启	数据库参数修改后，有些参数修改，需等待用户重启实例才能生效。

1.6 典型应用

1.6.1 读写分离

MySQL和PostgreSQL数据库的主实例和只读实例都具有独立的连接地址，每个云数据库MySQL、PostgreSQL单机实例、主备实例最多支持创建5个只读实例，创建方法请参见[创建只读实例](#)和[管理只读实例](#)。

用户只需通过添加只读实例的个数，即可不断扩展系统的处理能力，无需更改应用程序。

1.7 约束与限制

1.7.1 MySQL 约束与限制

关系型数据库服务上的MySQL在使用上有一些固定限制，用来提高实例的稳定性和安全性，具体详见[表1-7](#)。

表 1-7 功能约束与限制

功能	使用限制
数据库访问	<ul style="list-style-type: none">如果关系型数据库实例没开通公网访问，则该实例必须与弹性云服务器在同一个虚拟私有云内才能访问。关系型数据库只读实例必须创建在与主实例相同的一个子网内。弹性云服务器必须处于目标关系型数据库实例所属安全组允许访问的范围内。 如果关系型数据库实例与弹性云服务器处于不同的安全组，系统默认不能访问。需要在关系型数据库的安全组添加一条“入”的访问规则。MySQL实例的默认端口为3306，需用户后，才能访问其它端口。
部署	实例所部署的弹性云服务器，对用户都不可见，即只允许应用程序通过IP地址和端口访问数据库。
数据库的root权限	创建实例页面只提供管理员root用户权限。
修改数据库参数设置	大部分数据库参数可以通过控制台进行修改。
数据迁入	MySQL命令行工具方式迁入数据，具体请参考 使用mysqldump迁移MySQL数据 。
MySQL存储引擎	具体请参考 RDS for MySQL支持哪些存储引擎 。
搭建数据库复制	RDS for MySQL本身提供主备复制架构的双节点集群，无需用户手动搭建。其中主备复制架构集群的备实例不对用户开放，用户应用不可直接访问。
小版本升级	目前，RDS for MySQL实例最大可支持10万张表，大于10万张表时，可能会导致小版本升级失败。
重启RDS实例	无法通过命令行重启，必须通过关系型数据库服务的管理控制台操作重启实例。
查看RDS备份	具体请参考 下载备份文件 。

1.7.2 PostgreSQL 约束与限制

关系型数据库服务上的PostgreSQL在使用上有一些固定限制，用来提高实例的稳定性和安全性，具体详见[表1-8](#)。

表 1-8 功能约束与限制

功能	使用限制
数据库访问	<ul style="list-style-type: none">如果关系型数据库实例未开通公网访问，则该实例必须与弹性云服务器处在同一个虚拟私有云内才能相互访问。关系型数据库只读实例必须创建在与主实例相同的子网内。弹性云服务器必须处于目标关系型数据库实例所属安全组允许访问的范围内。 如果关系型数据库实例与弹性云服务器处于不同的安全组，系统默认不能访问。需要在关系型数据库的安全组添加一条“入”的访问规则。关系型数据库实例的默认端口为5432，需用户手动修改端口号后，才能访问其他端口。
部署	实例所部署的弹性云服务器，对用户都不可见，即只允许应用程序通过IP地址和端口访问数据库。
数据库的root权限	创建实例页面只提供管理员root用户权限。
修改数据库参数设置	大部分数据库参数可以通过控制台进行修改。
数据迁入	使用psql命令行工具方式迁入数据，具体请参考 使用psql命令迁移PostgreSQL数据 。
搭建数据库复制	RDS for PostgreSQL本身提供主备复制架构的双节点集群，无需用户手动搭建。其中主备复制架构集群的备实例不对用户开放，用户应用不可直接访问。
重启RDS实例	无法通过命令行重启，必须通过关系型数据库服务的管理控制台操作重启实例。
RDS备份查看	具体请参考 下载全量备份文件 。

1.7.3 Microsoft SQL Server 约束与限制

SQL Server引擎的关系型数据库服务仅提供附带许可实例，即实例创建后具有微软SQL Server软件对应版本许可授权，不提供用户自带许可。

为保障用户在使用上的安全性，RDS for SQL Server的部分功能存在限制，详见[表 1-9](#)。

RDS for SQL Server实例分为三个实例类型，即单机实例、主备实例和集群版实例。不同系列支持的功能不同，

表 1-9 功能和约束与限制

功能	单机实例	主备实例
数据库数量	100	100

功能	单机实例	主备实例
数据库账号数量	无限制	无限制
创建用户、LOGIN或数据库	支持	支持
数据库级别的DDL触发器	支持	支持
数据库内的授权权限	支持	支持
杀线程权限	支持	支持
SQL Profiler	支持	支持
优化顾问	支持	支持
变更数据捕获（CDC）	支持	支持
变更跟踪	支持	支持
Windows域账号登录	支持	支持
数据集成功能（SSIS）	不支持	不支持
数据分析功能（SSAS）	不支持	不支持
R语言服务	不支持	不支持
异步消息通讯	不支持	不支持
复制订阅功能	不支持	不支持
策略管理	不支持	不支持

1.8 RDS 与其他服务的关系

表 1-10 与其他服务的关系

相关服务	交互功能
弹性云服务器（ECS）	关系型数据库服务配合弹性云服务器（Elastic Cloud Server，简称ECS）一起使用，通过内网连接关系型数据库可以有效地降低应用响应时间、节省公网流量费用。
虚拟私有云（VPC）	对您的关系型数据库实例进行网络隔离和访问控制。
对象存储服务（OBS）	存储关系型数据库实例的自动和手动备份数据。

2 MySQL 快速入门

2.1 操作指引

本章节提供了如何在管理控制台创建实例、连接实例的具体操作流程。

流程图

图 2-1 操作流程图



操作步骤

表 2-1 相关操作及参考手册

相关操作	参考手册
创建RDS实例	步骤一：创建实例
连接RDS实例	步骤二：连接实例

2.2 步骤一：创建实例

操作场景

本节将介绍在关系型数据库服务的管理控制台创建实例的过程。

您可以根据业务需要定制相应计算能力和存储空间的关系型数据库实例。

操作步骤

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 在“实例管理”页面，单击“创建数据库实例”。

步骤4 在“服务选型”页面，填写并选择实例相关信息后，单击“立即创建”。

表 2-2 基本信息

参数	描述
区域	租户当前所在区域，也可在页面左上角切换。 说明 不同区域内的产品内网不互通，且创建后不能更换，请谨慎选择。
实例名称	实例名称长度在4个到64个字符之间，必须以字母开头，可以包含字母、数字、中划线、下划线，不能包含其他特殊字符。
数据库引擎	MySQL。
数据库版本	请参见 数据库引擎和版本 。 不同区域所支持的数据库版本不同，请以实际界面为准。 选用MySQL数据库时，请根据实际业务需求选择合适的数据库引擎版本。建议您选择当前可用的最高版本数据库，因其性能更稳定，安全性更高，使用更可靠。
实例类型+可用区	<ul style="list-style-type: none">主备：一主一备的经典高可用架构。适用于大中型企业的生产数据库，覆盖互联网、物联网、零售电商、物流、游戏等行业应用。备机提高了实例的可靠性，创建主机的过程中，同步创建备机，备机创建成功后，用户不可见。 可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。 RDS支持在同一个可用区内或者跨可用区部署数据库主备实例，以提供故障切换能力和高可用性。单机：采用单个数据库节点部署架构，与主流的主备实例相比，它只包含一个节点，但具有高性价比。适用于个人学习、微型网站以及中小企业的开发测试环境。

参数	描述
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s。超高I/O：最大吞吐量350MB/s。
时区	由于世界各国与地区经度不同，地方时也有所不同，因此会划分为不同的时区。时区可在创建实例时选择，后期可修改。

表 2-3 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。 关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。 数据库创建成功后可进行扩容，具体请参见 磁盘扩容 。 说明 <ul style="list-style-type: none">存储空间支持40GB到4000GB，用户选择容量大小必须为10的整数倍。

表 2-4 网络

参数	描述
虚拟私有云	关系型数据库实例所在的虚拟专用网络，可以对不同业务进行网络隔离。您需要创建或选择所需的虚拟私有云。如何创建虚拟私有云，请参见《虚拟私有云用户指南》中的“创建虚拟私有云基本信息及默认子网”。 如果没有可选的虚拟私有云，关系型数据库服务默认为您分配资源。 须知 目前RDS实例创建完成后不支持切换虚拟私有云，请谨慎选择所属虚拟私有云。

参数	描述
子网	<p>通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全性。子网在可用区内才会有效，创建关系型数据库实例的子网默认开启DHCP功能，不可关闭。</p> <p>创建实例时RDS会自动为您配置内网地址，您也可输入子网号段内未使用的内网地址，实例创建成功后该内网地址可修改。</p>
安全组	<p>安全组限制实例的安全访问规则，加强关系型数据库服务与其他服务间的安全访问。请确保所选取的安全组允许客户端访问数据库实例。</p> <p>如果不创建安全组或没有可选的安全组，关系型数据库服务默认为您分配安全组资源。</p>

表 2-5 数据库配置

参数	描述
管理员帐户	数据库的登录名默认为root。
管理员密码	<p>所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*_=-+?,特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。</p> <p>请妥善保管您的密码，因为系统将无法获取您的密码信息。</p> <p>实例创建成功后，如需重置密码，请参见重置管理员密码。</p>
确认密码	必须和管理员密码相同。
参数模板	<p>数据库参数模板就像是数据库引擎配置值的容器，参数模板中的参数可应用于一个或多个相同类型的数据库实例。对于HA实例创建成功后，主备参数模板相同。实例创建成功后，参数模板可进行修改。</p> <p>须知</p> <p>创建数据库实例时，为确保数据库实例正常创建，自定义参数模板中相关规格参数如下不会下发，而是采用系统默认的推荐值。</p> <ul style="list-style-type: none">• “back_log”• “innodb_io_capacity_max”• “max_connections”• “innodb_io_capacity”• “innodb_buffer_pool_size”• “innodb_buffer_pool_instances” <p>您可以在实例创建完成之后根据业务需要进行调整。具体请参见编辑参数模板。</p>

参数	描述
企业项目	对于已成功关联企业项目的用户，仅需在“企业项目”下拉框中选择目标项目。 如果需要自定义企业项目，请前往项目管理服务进行创建。关于如何创建项目，详见《项目管理用户指南》。

表 2-6 批量创建

参数	描述
创建数量	关系型数据库服务支持批量创建实例，如果您选择创建主备实例，数量选择为1，那么会同步创建一个主实例和一个备实例。 创建多个数据库实例时，名称自动按序增加4位数字后缀。例如输入instance，从instance-0001开始命名；若已有instance-0010，从instance-0011开始命名。

表 2-7 创建只读实例

参数	描述
只读实例名称	只读实例名称默认在主实例名称后增加后缀read和2位数字。例如主实例名称为instance-0001，只读实例从instance-0001-read-01开始命名。 只读实例的网络、存储配置与主实例一致。
只读实例可用区	默认和主实例跨AZ，您也可根据业务需要进行选择。 须知 不同区域内的产品内网不互通，且购买后不能更换，请谨慎选择。
性能规格	只读实例的CPU和内存。
只读实例数量	关系型数据库服务支持在下发创建主实例任务时，同步下发批量创建只读实例的任务，待主实例创建成功后，系统自动触发创建只读实例，您最多可在一个主实例下创建5个只读实例。 如果您选择创建主备实例，数量选择为1，那么会创建一个主备实例和一个只读实例。

📖 说明

关系型数据库的性能，取决于用户申请关系型数据库时所选择的配置。可供用户选择的硬件配置项为性能规格、存储类型以及存储空间。

步骤5 进行规格确认。

- 如果需要重新选择实例规格，单击“上一步”，回到上个页面修改实例信息。
- 如果规格确认无误，单击“提交”，完成创建实例的申请。

步骤6 关系型数据库实例创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

- 创建实例过程中，状态显示为“创建中”，创建完成的实例状态显示为“正常”。您可以通过“任务中心”查看详细进度和结果。
- 创建实例时，系统默认开启自动备份策略，后期可修改。实例创建成功后，系统会自动创建一个全量备份。
- 数据库端口默认为3306，实例创建成功后可修改。

具体请参见[修改数据库端口](#)。

---结束

2.3 步骤二：连接实例

2.3.1 实例连接方式介绍

关系型数据库服务提供使用内网、公网的连接方式。

表 2-8 RDS 连接方式

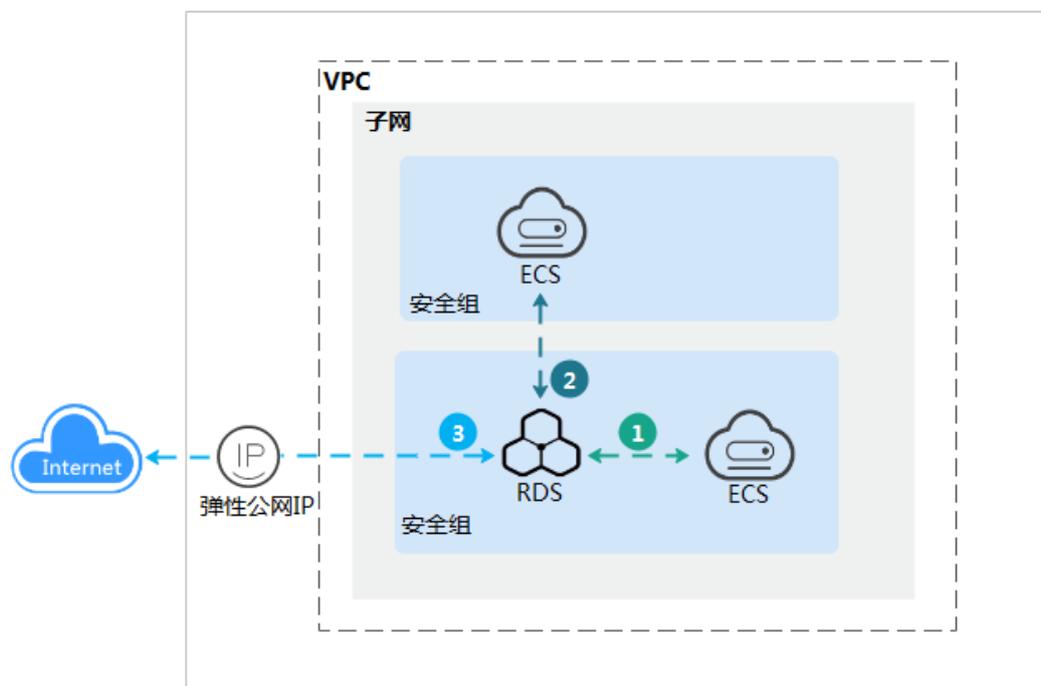
连接方式	IP地址	使用场景	说明
内网连接	内网IP地址	系统默认提供内网IP地址。 当应用部署在弹性云服务器上，且该弹性云服务器与关系型数据库实例处于同一区域，同一VPC时，建议单独使用内网IP连接弹性云服务器与关系型数据库实例。	<ul style="list-style-type: none">• 安全性高，可实现RDS的较好性能。• 推荐使用内网连接。
公网连接	弹性公网IP	不能通过内网IP地址访问RDS实例时，使用公网访问，建议单独绑定弹性公网IP连接弹性云服务器（或公网主机）与关系型数据库实例。	<ul style="list-style-type: none">• 安全性低。• 为了获得更快的传输速率和更高的安全性，建议您将应用迁移到与您的RDS实例在同一VPC内，使用内网连接。

说明

- VPC：虚拟私有云（Virtual Private Cloud，简称VPC）。
- ECS：弹性云服务器（Elastic Cloud Server，简称ECS）。
- 若弹性云服务器和关系型数据库实例处于同一个VPC内，则无需申请外网地址。

其中，通过内网和公网的连接方式如[图2-2](#)所示。

图 2-2 实例内网和公网连接



- 1 通过内网连接RDS实例（ECS与RDS在相同安全组）
- 2 通过内网连接RDS实例（ECS与RDS在不同安全组）
- 3 通过公网连接RDS实例

2.3.2 通过内网连接 MySQL 实例

2.3.2.1 内网连接实例流程

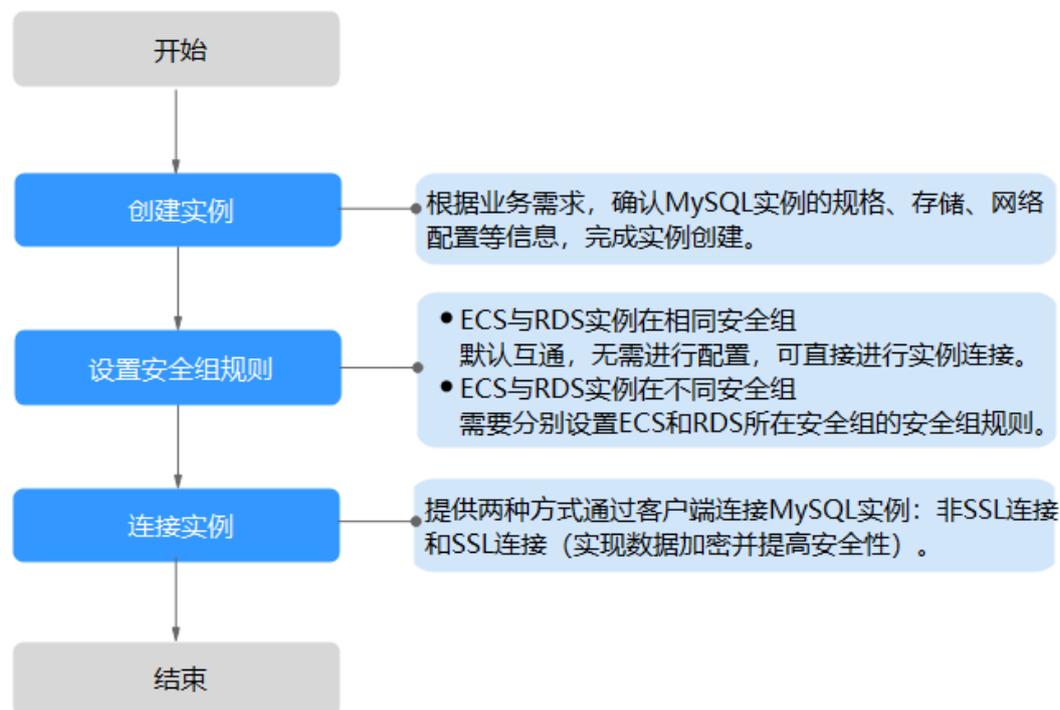
操作场景

首次使用关系型数据库服务，请先了解[MySQL约束与限制](#)。

使用流程

通过内网连接MySQL实例的使用流程介绍如[图2-3](#)所示。

图 2-3 通过内网连接实例



2.3.2.2 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

内网连接RDS实例时，设置安全组分为以下两种情况：

- ECS与RDS实例在相同安全组时，默认ECS与RDS实例互通，无需设置安全组规则，执行[通过内网连接MySQL实例](#)。
- ECS与RDS实例在不同安全组时，需要为RDS和ECS分别设置安全组规则。
 - 设置RDS安全组规则：为RDS所在安全组配置相应的**入方向规则**。
 - 设置ECS安全组规则：安全组默认规则为出方向上数据报文全部放行，此时，无需对ECS配置安全组规则。当在ECS所在安全组为非默认安全组且出方向规则**非全放通**时，需要为ECS所在安全组配置相应的**出方向规则**。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建100个安全组。
- 默认情况下，一个安全组最多只允许拥有50条安全组规则。

- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的入方向规则。

📖 说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

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 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤4 设置安全组规则。

- 在“基本信息”页面中，在“连接信息”模块的“安全组”处，单击安全组名称，进入安全组页面。

图 2-4 连接信息



步骤5 在入方向规则页签，单击“添加规则”，添加入方向规则。

单击“+”可以依次增加多条入方向规则。

图 2-5 添加入方向规则



表 2-9 入方向参数说明

参数	说明	取值样例
协议端口	网络协议。目前支持“All”、“TCP”、“UDP”、“ICMP”和“GRE”等协议。	Custom TCP
	端口：允许远端地址访问弹性云服务器指定端口。	通过内网连接实例时 ，输入已购买的目标实例的端口。
源地址	源地址：可以是IP地址、安全组。例如： <ul style="list-style-type: none">• xxx.xxx.xxx.xxx/32（IPv4地址）• xxx.xxx.xxx.0/24（子网）• 0.0.0.0/0（任意地址）	0.0.0.0/0
描述	安全组规则的描述信息，非必填项。 描述信息内容不能超过255个字符，且不能包含“<”和“>”。	-

----结束

2.3.2.3 通过内网连接 MySQL 实例

提供两种连接方式通过MySQL客户端连接实例：非SSL连接和SSL连接。其中，SSL连接实现了数据加密功能，具有更高的安全性。

前提条件

1. 登录弹性云服务器。

- 通过弹性云服务器连接关系型数据库实例，需要具备以下条件。
 - 该弹性云服务器与目标实例必须处于同一VPC内。
 - 该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。
 - 如果目标实例所属安全组为**默认安全组**，则无需设置安全组规则。
 - 如果目标实例所属安全组**非默认安全组**，请查看安全组规则是否允许该弹性云服务器访问。具体操作请参考[设置安全组规则](#)。
如果安全组规则允许弹性云服务器访问，即可连接实例。
如果安全组规则不允许弹性云服务器访问，则需添加安全组规则。
该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。

2. 使用客户端连接实例。

您可以在Linux操作系统和Windows操作系统中，使用数据库客户端连接RDS实例。

- 在Linux操作系统中，您需要在可访问关系型数据库的设备上安装MySQL客户端。建议您下载的MySQL客户端版本高于已创建的RDS实例中数据库版本。如何获取相应安装包及完成安装，请参见[如何安装MySQL客户端](#)。

- 在Windows操作系统中，您可以使用任何通用的数据库客户端连接到RDS实例且连接方法类似。

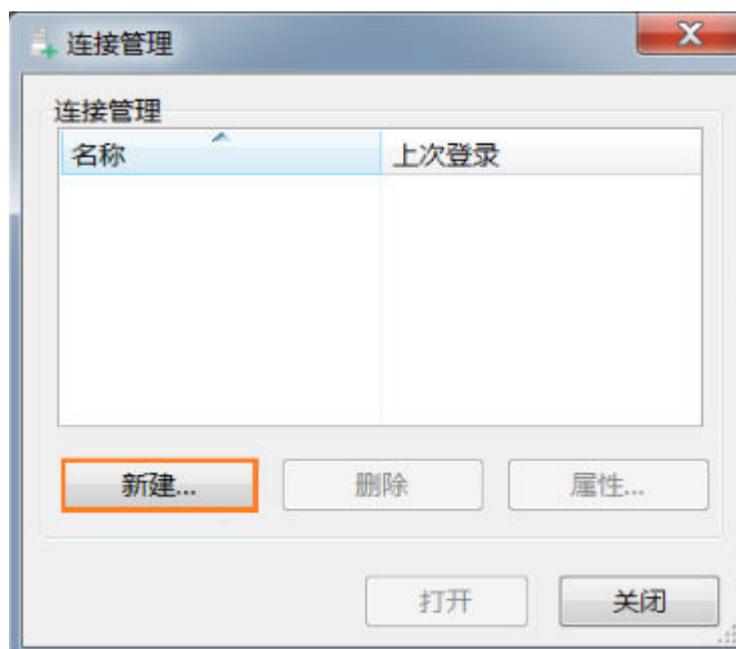
以MySQL-Front为例，具体操作请参见[使用MySQL-Front连接实例](#)。

使用 MySQL-Front 连接实例

步骤1 启动MySQL-Front客户端。

步骤2 在连接管理对话框中，单击“新建”。

图 2-6 连接管理



步骤3 输入需要连接的关系型数据库实例信息，然后单击“确定”。如[图2-7](#)所示。

图 2-7 添加信息

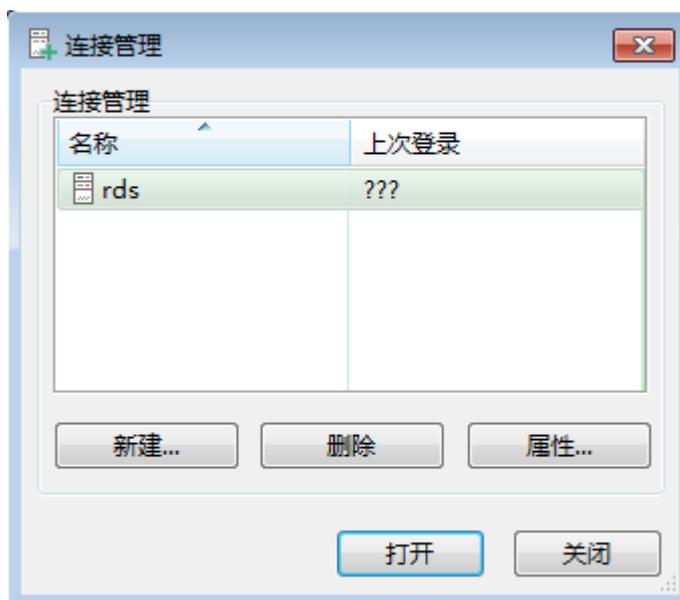


表 2-10 参数说明

参数	说明
名称	连接数据库的任务名称。若不填写，系统默认与Host一致。
主机	目标实例的内网地址。查看目标实例的内网地址及端口信息的步骤如下： 1. 登录关系型数据库的管理控制台。 2. 选择目标实例所在区域。 3. 单击目标实例名称，进入“基本信息”页面。 4. 在“连接信息”模块，可查看“内网地址”信息。
端口	输入RDS实例的内网端口。
用户	需要访问RDS实例的账号名称。默认root。
密码	要访问关系型数据库实例的账号所对应的密码。

步骤4 在“连接管理”窗口，选中**步骤3**创建的连接，单击“打开”，如下图所示。若连接信息无误，即会成功连接实例。

图 2-8 打开登录信息



说明

若连接失败，请参见[内网方式下ECS无法连接RDS实例的原因](#)。

----结束

2.3.3 通过公网连接 MySQL 实例

2.3.3.1 公网连接实例流程

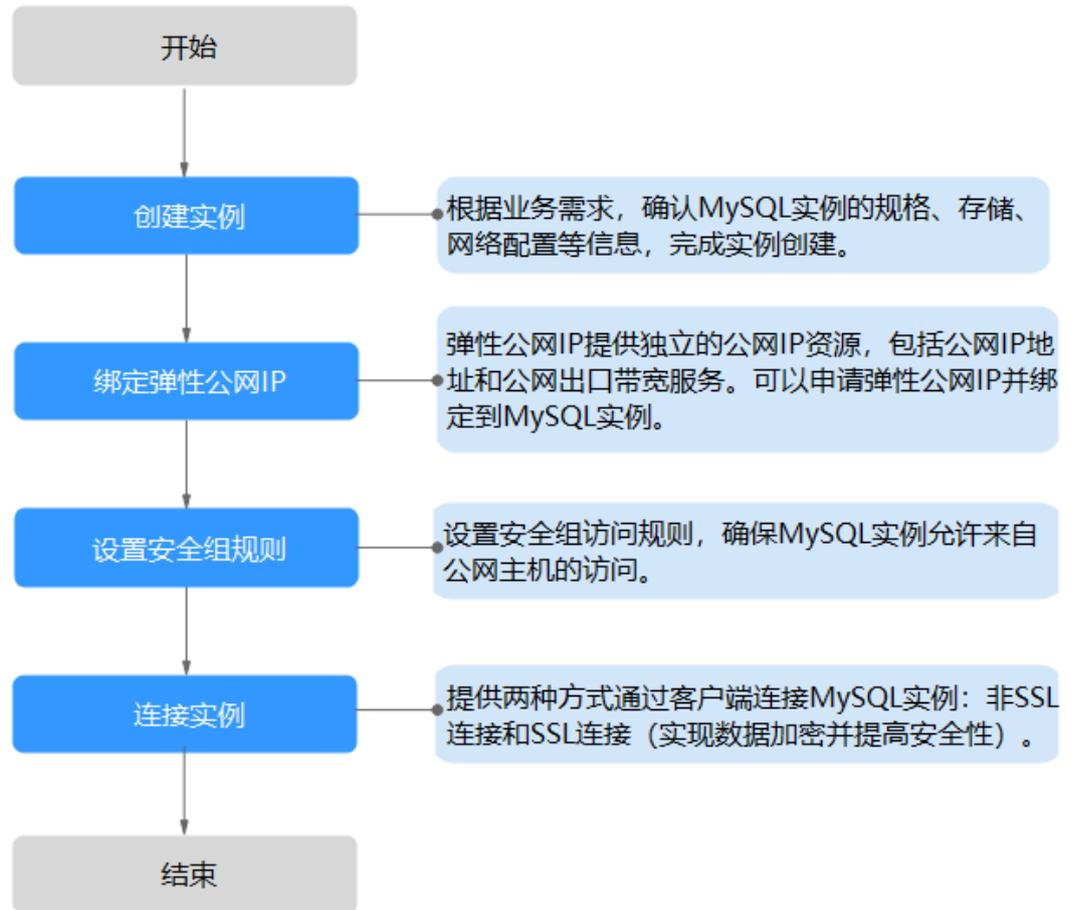
操作场景

首次使用关系型数据库服务，请先了解[MySQL约束与限制](#)。

使用流程

通过公网连接MySQL实例的使用流程介绍如[图2-9](#)所示。

图 2-9 通过公网连接实例



2.3.3.2 绑定弹性公网 IP

操作场景

关系型数据库实例创建成功后，默认未开启公网访问功能（即未绑定弹性公网IP）。关系型数据库服务支持用户绑定弹性公网IP，在公共网络来访问数据库实例，绑定后也可根据需要解绑。

注意事项

- 您需要设置安全组，开通需访问数据库的IP地址和端口，才可以访问数据库实例。在访问数据库前，您需要将访问数据库的IP地址，或者IP段加安全组入方向的访问规则，操作请参见[设置安全组规则](#)。
- 公网访问会降低实例的安全性，请谨慎选择。为了获得更快的传输速率和更高的安全级别，建议您将应用迁移到与您的关系型数据库在同一区域的弹性云服务器上。

绑定弹性公网 IP

步骤1 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤2 在左侧导航栏，单击“弹性公网IP”，单击“绑定弹性公网IP”。

步骤3 在弹出框的弹性公网IP地址列表中，选择目标弹性公网IP，单击“确定”，提交绑定任务。

如果没有可用的弹性公网IP，您可单击“查看弹性公网IP”，获取弹性公网IP。

步骤4 查看绑定成功的弹性公网IP。

您也可以在“任务中心”页面，查看“实例绑定弹性公网IP”任务的执行进度及结果。

---结束

2.3.3.3 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

通过弹性公网IP连接RDS实例时，需要为RDS所在安全组配置相应的入方向规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建100个安全组。
- 默认情况下，一个安全组最多只允许拥有50条安全组规则。
- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的入方向规则。

说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

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 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤4 设置安全组规则。

- 在“基本信息”页面中，在“连接信息”模块的“安全组”处，单击安全组名称，进入安全组页面。

图 2-10 连接信息



步骤5 在入方向规则页签，单击“添加规则”，添加入方向规则。
单击“+”可以依次增加多条入方向规则。

图 2-11 添加入方向规则



表 2-11 入方向参数说明

参数	说明	取值样例
协议端口	网络协议。目前支持“ALL”、“TCP”、“UDP”、“ICMP”和“GRE”等协议。	Custom TCP
	端口：允许远端地址访问弹性云服务器指定端口。	通过公网连接实例时 ，输入已购买的目标实例的端口。
源地址	源地址：可以是IP地址、安全组。 例如： <ul style="list-style-type: none"> xxx.xxx.xxx.xxx/32（IPv4地址） xxx.xxx.xxx.0/24（子网） 0.0.0.0/0（任意地址） 	0.0.0.0/0

参数	说明	取值样例
描述	安全组规则的描述信息，非必填项。 描述信息内容不能超过255个字符，且不能包含“<”和“>”。	-

----结束

2.3.3.4 通过公网连接 MySQL 实例

提供两种连接方式通过MySQL客户端连接实例：非SSL连接和SSL连接。其中，SSL连接实现了数据加密功能，具有更高的安全性。

前提条件

1. **绑定弹性公网IP并设置安全组规则。**
 - a. 对目标实例绑定弹性公网IP。
关于如何绑定弹性公网IP，请参见[绑定弹性公网IP](#)。
 - b. 获取本地设备的IP地址。
 - c. 设置安全组规则。
将**1.b**中获取的IP地址及目标实例的端口加入安全组允许访问的范围中。
关于如何设置安全组规则，请参见[设置安全组规则](#)章节。
 - d. 使用ping命令连通**1.a**中绑定的弹性公网IP，确保本地设备可以访问该弹性公网IP。
2. **使用客户端连接实例。**

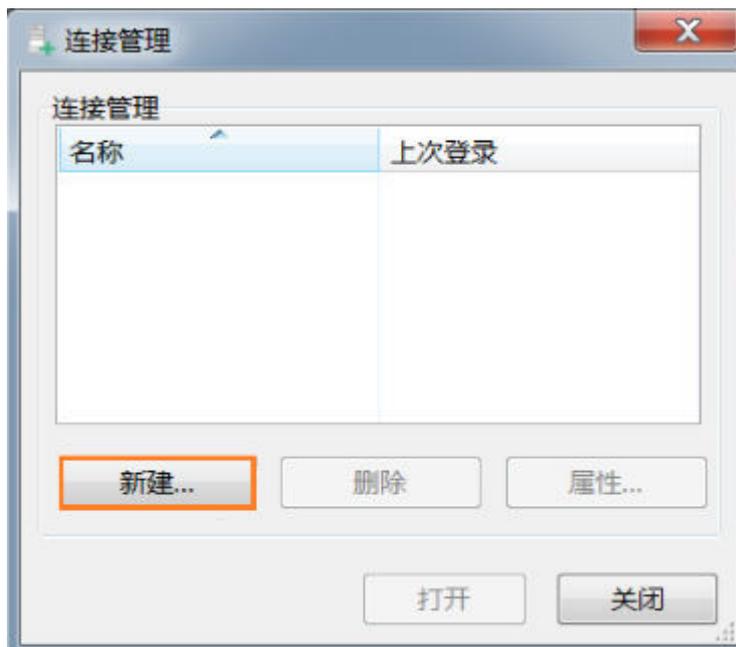
您可以在Linux操作系统和Windows操作系统中，使用数据库客户端连接RDS实例。

 - 在Linux操作系统中，您需要在弹性云服务器上安装MySQL客户端。建议您下载的MySQL客户端版本高于已创建的RDS实例中数据库版本。
如何获取相应安装包及完成安装，请参见[如何安装MySQL客户端](#)。
 - 在Windows操作系统中，您可以使用任何通用的数据库客户端连接到RDS实例且连接方法类似。
以MySQL-Front为例，具体操作请参见[使用MySQL-Front连接实例](#)。

使用 MySQL-Front 连接实例

- 步骤1 启动MySQL-Front客户端。
- 步骤2 在连接管理对话框中，单击“新建”。

图 2-12 连接管理



步骤3 输入需要连接的关系型数据库实例信息，然后单击“确定”。如图2-13所示。

图 2-13 添加信息

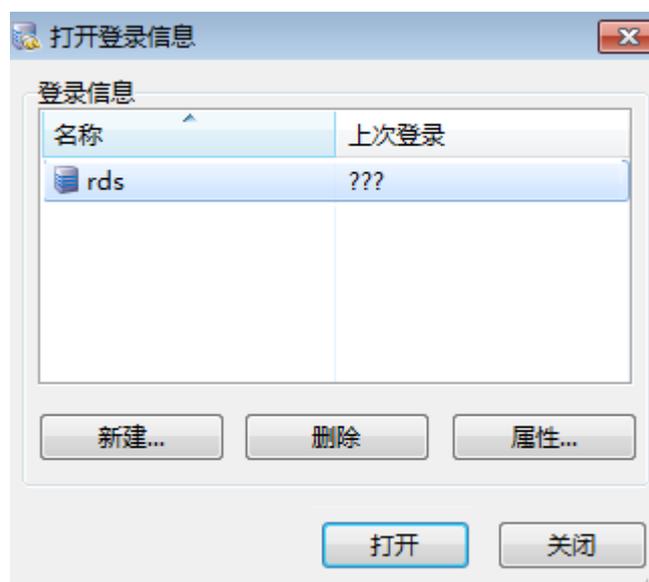


表 2-12 参数说明

参数	说明
名称	连接数据库的任务名称。若不填写，系统默认与Host一致。
主机	目标实例的弹性公网IP。
端口	输入RDS实例的内网端口。
用户	需要访问RDS实例的账号名称。默认root。
密码	要访问关系型数据库实例的账号所对应的密码。

步骤4 在打开登录信息窗口，选中图2-14创建的连接，单击“打开”，如下图所示。
若连接信息无误，即会成功连接实例。

图 2-14 打开登录信息



说明

若连接失败，请确保各项前提条件正确配置后，重新尝试连接。

----结束

3 PostgreSQL 快速入门

3.1 操作指引

本章节提供了如何在管理控制台创建实例、连接实例的具体操作流程。

流程图

图 3-1 操作流程图



操作步骤

表 3-1 相关操作及参考手册

相关操作	参考手册
创建RDS实例	步骤一：创建实例
连接RDS实例	步骤二：连接实例

3.2 步骤一：创建实例

操作场景

本文将介绍在关系型数据库服务的管理控制台创建实例的过程。
您可以根据业务需要定制相应计算能力和存储空间的RDS实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击“创建数据库实例”。

步骤5 在“服务选型”页面，填写并选择实例相关信息后，单击“立即创建”。

表 3-2 基本信息

参数	描述
区域	租户当前所在区域，也可在页面左上角切换。 说明 不同区域内的产品内网不互通，且创建后不能更换，请谨慎选择。
实例名称	实例名称长度在4个到64个字符之间，必须以字母开头，可以包含字母、数字、中划线、下划线或中文，不能包含其他特殊字符。
数据库引擎	PostgreSQL。
数据库版本	请参见 数据库引擎和版本 。 不同区域所支持的数据库版本不同，请以实际界面为准。 选用PostgreSQL数据库时，请根据实际业务需求选择合适的数据库引擎版本。建议您选择当前可用的最高版本数据库，因其性能更稳定，安全性更高，使用更可靠。
实例类型+可用区	<ul style="list-style-type: none">主备：一主一备的经典高可用架构。适用于大中型企业的生产数据库，覆盖互联网、物联网、零售电商、物流、游戏等行业应用。备机提高了实例的可靠性，创建主机的过程中，同步创建备机，备机创建成功后，用户不可见。可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。 RDS支持在同一个可用区内或者跨可用区部署数据库主备实例，以提供故障切换能力和高可用性。单机：采用单个数据库节点部署架构，与主流的主备实例相比，它只包含一个节点，但具有高性价比。适用于个人学习、微型网站以及中小企业的开发测试环境。

参数	描述
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s超高I/O：最大吞吐量350MB/s
时区	由于世界各国家与地区经度不同，地方时也有所不同，因此会划分为不同的时区。时区可在创建实例时选择，后期可修改。

表 3-3 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。存储空间支持40GB到4000GB，用户选择容量大小必须为10的整数倍。 创建成功后可进行扩容，具体请参见 磁盘扩容 。

表 3-4 网络

参数	描述
虚拟私有云	关系型数据库实例所在的虚拟专用网络，可以对不同业务进行网络隔离。您需要创建或选择所需的虚拟私有云。如何创建虚拟私有云，请参见《 虚拟私有云用户指南 》中的“ 创建虚拟私有云基本信息及默认子网 ”。 如果没有可选的虚拟私有云，关系型数据库服务默认为您分配资源。 须知 目前RDS实例创建完成后不支持切换虚拟私有云，请谨慎选择所属虚拟私有云。

参数	描述
子网	<p>通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全性。子网在可用区内才会有效，创建关系型数据库实例的子网默认开启DHCP功能，不可关闭。</p> <p>创建实例时RDS会自动为您配置内网地址，您也可输入子网号段内未使用的内网地址，实例创建成功后该内网地址可修改。</p>
安全组	<p>控制网络出/入及端口的访问，默认添加了关系型数据库实例所属的安全组访问。</p> <p>安全组限制实例的安全访问规则，加强关系型数据库服务与其他服务间的安全访问。请确保所选取的安全组允许客户端访问数据库实例。</p> <p>如果没有可选的安全组，关系型数据库服务默认为您分配安全组资源。</p>

表 3-5 数据库配置

参数	描述
管理员账户	数据库的登录名称默认为root。
管理员密码	<p>所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%^*_-=+?,特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。</p> <p>请妥善保管您的密码，因为系统将无法获取您的密码信息。</p> <p>实例创建成功后，如需重置密码，请参见重置管理员密码。</p>
确认密码	必须和主密码相同。
参数模板	<p>数据库参数模板就像是数据库引擎配置值的容器，参数模板中的参数可应用于一个或多个相同类型的数据库实例。对于HA实例创建成功后，主备参数模板相同。实例创建成功后，参数模板可进行修改。</p> <p>须知</p> <p>创建数据库实例时，为确保数据库实例正常创建，自定义参数模板中相关规格参数如下不会下发，而是采用系统默认的推荐值。</p> <ul style="list-style-type: none">“maintenance_work_mem”“shared_buffers”“max_connections”“effective_cache_size” <p>您可以在实例创建完成之后根据业务需要进行调整。具体请参见修改实例参数。</p>
企业项目	<p>对于已成功关联企业项目的用户，仅需在“企业项目”下拉框中选择目标项目。</p> <p>如果需要自定义企业项目，请前往项目管理服务进行创建。关于如何创建项目，详见《项目管理用户指南》。</p>

表 3-6 批量创建

参数	描述
创建数量	关系型数据库服务支持批量创建实例，如果您选择创建主备实例，数量选择为1，那么会同步创建一个主实例和一个备实例。 创建多个数据库实例时，名称自动按序增加4位数字后缀。例如输入instance，从instance-0001开始命名；若已有instance-0010，从instance-0011开始命名。

📖 说明

关系型数据库的性能，取决于用户申请关系型数据库时所选择的配置。可供用户选择的硬件配置项为性能规格、存储类型以及存储空间。

步骤6 进行规格确认。

- 如果需要重新选择实例规格，单击“上一步”，回到上个页面修改关系型数据库实例信息。
- 如果规格确认无误，单击“提交”，完成关系型数据库实例的申请。

步骤7 关系型数据库实例创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

- 创建实例过程中，状态显示为“创建中”，创建完成的实例状态显示为“正常”。
- 创建关系型数据库实例时，系统默认开启自动备份策略，后期可修改。实例创建成功后，系统会自动创建一个全量备份。
- 数据库端口默认为5432，实例创建成功后可修改。

具体请参见[修改数据库端口](#)。

----结束

3.3 步骤二：连接实例

3.3.1 实例连接方式介绍

关系型数据库服务提供使用内网、公网的连接方式。

表 3-7 RDS 连接方式

连接方式	IP地址	使用场景	说明
内网连接	内网IP地址	系统默认提供内网IP地址。 当应用部署在弹性云服务器上，且该弹性云服务器与关系型数据库实例处于同一区域，同一VPC时，建议单独使用内网IP连接弹性云服务器与关系型数据库实例。	<ul style="list-style-type: none">• 安全性高，可实现RDS的较好性能。• 推荐使用内网连接。

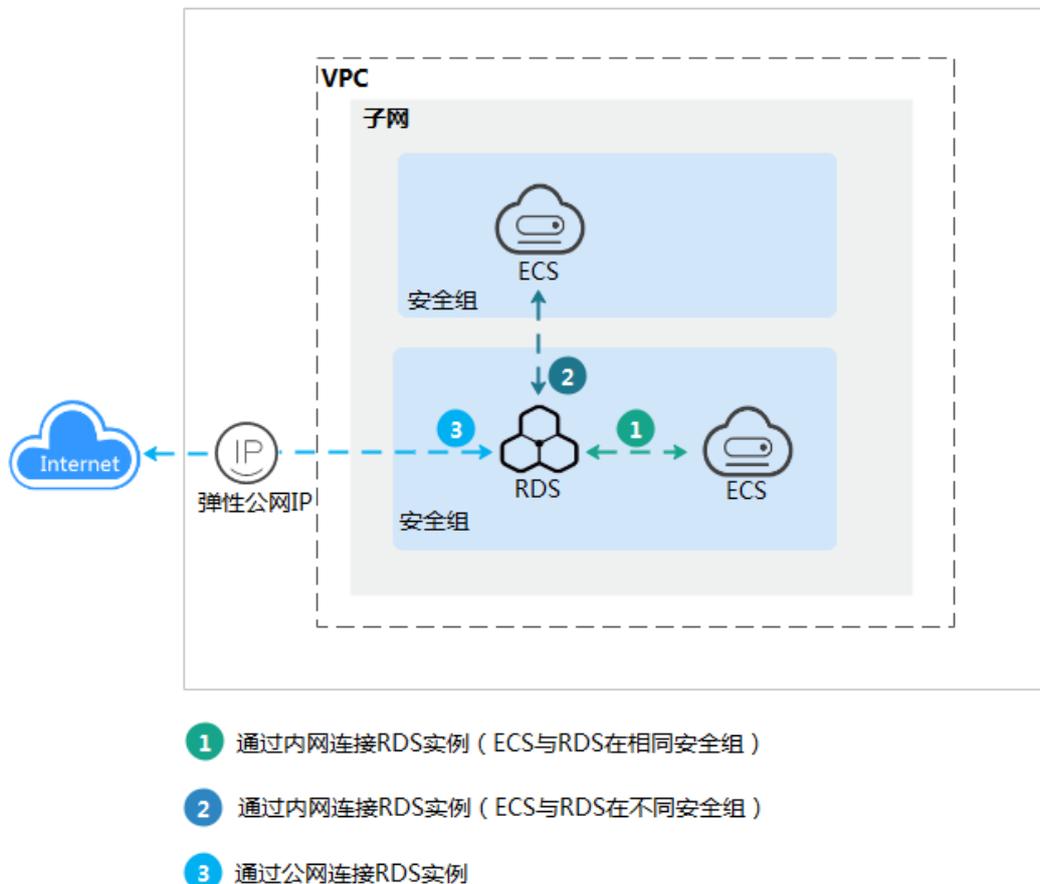
连接方式	IP地址	使用场景	说明
公网连接	弹性公网IP	不能通过内网IP地址访问RDS实例时，使用公网访问，建议单独绑定弹性公网IP连接弹性云服务器（或公网主机）与关系型数据库实例。	<ul style="list-style-type: none"> 降低安全性。 为了获得更快的传输速率和更高的安全性，建议您将应用迁移到与您的RDS实例在同一VPC内，使用内网连接。

说明

- VPC：虚拟私有云（Virtual Private Cloud，简称VPC）。
- ECS：弹性云服务器（Elastic Cloud Server，简称ECS）。
- 若弹性云服务器和关系型数据库实例处于同一个VPC内，则无需申请外网地址。

其中，通过内网和公网的连接方式如图3-2所示。

图 3-2 实例内网和公网连接



3.3.2 通过内网连接 PostgreSQL 实例

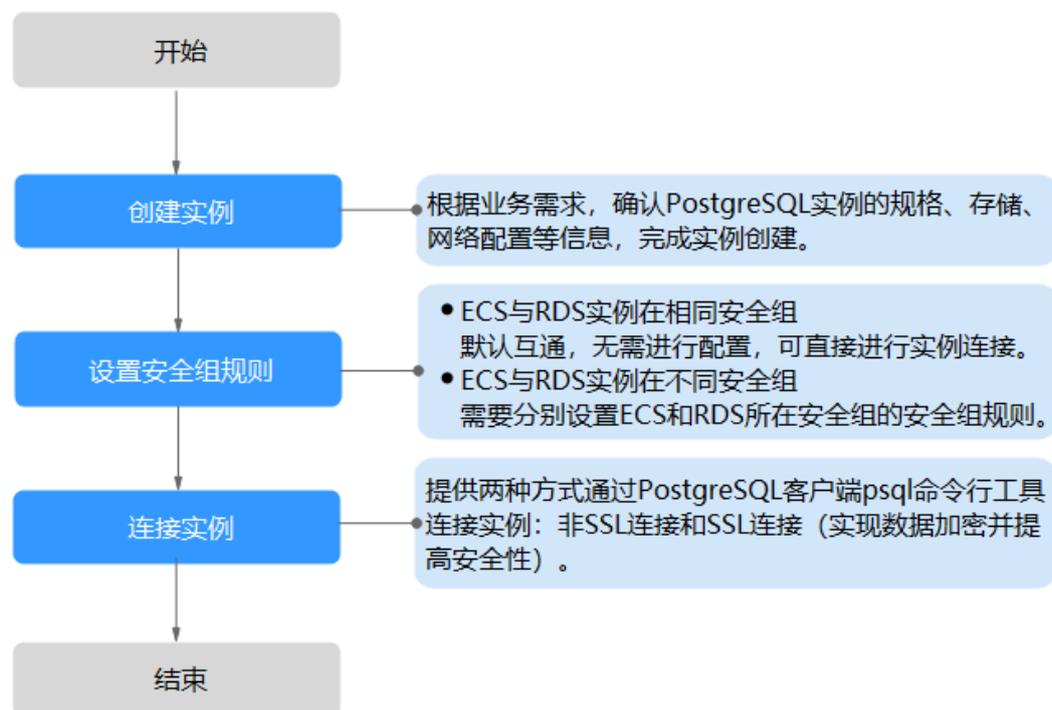
3.3.2.1 内网连接实例流程

首次使用关系型数据库服务，请先了解[PostgreSQL约束与限制](#)。

使用流程

通过内网连接PostgreSQL实例的使用流程介绍如[图3-3](#)所示。

图 3-3 通过内网连接实例



3.3.2.2 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

内网连接RDS实例时，设置安全组分为以下两种情况：

- ECS与RDS实例在相同安全组时，默认ECS与RDS实例互通，无需设置安全组规则，执行[通过psql连接实例](#)。
- ECS与RDS实例在不同安全组时，需要为RDS和ECS分别设置安全组规则。
 - 设置RDS安全组规则：为RDS所在安全组配置相应的入方向规则。

- 设置ECS安全组规则：安全组默认规则为出方向上数据报文全部放行，此时，无需对ECS配置安全组规则。当在ECS所在安全组为非默认安全组且出方向规则**非全放通**时，需要为ECS所在安全组配置相应的**出方向**规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建100个安全组。
- 默认情况下，一个安全组最多只允许拥有50条安全组规则。
- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的**入方向**规则。

📖 说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

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 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤4 设置安全组规则。

在“连接信息”模块的“安全组”处，单击安全组名称，进入安全组页面。

图 3-4 连接信息



连接信息			
内网地址	10.0.1.63  修改	虚拟私有云	vpc02
数据库端口	5432  	子网	subnet-1 (10.0.1.0/24)
建议最大连接数	300	安全组	asddsas-cce-node-p0jcl 

步骤5 在入方向规则页签，单击“添加规则”，添加入方向规则。

单击“+”可以依次增加多条入方向规则。

图 3-5 添加入方向规则



表 3-8 入方向参数说明

参数	说明	取值样例
协议端口	网络协议。目前支持“All”、“TCP”、“UDP”、“ICMP”和“GRE”等协议。	Custom TCP
	端口：允许远端地址访问弹性云服务器指定端口，取值范围为：1～65535。	通过内网连接实例时 ，输入已购买的弹性云服务器的目标实例的端口。
源地址	源地址：可以是IP地址、安全组。 例如： <ul style="list-style-type: none"> xxx.xxx.xxx.xxx/32（IPv4地址） xxx.xxx.xxx.0/24（子网） 0.0.0.0/0（任意地址） 	0.0.0.0/0
描述	安全组规则的描述信息，非必填项。 描述信息内容不能超过255个字符，且不能包含“<”和“>”。	-

----结束

3.3.2.3 通过 psql 连接实例

通过PostgreSQL客户端连接实例的方式有非SSL连接和SSL连接两种，其中SSL连接通过了加密功能，具有更高的安全性。

前提条件

1. 登录弹性云服务器。

- 通过弹性云服务器连接关系型数据库实例，需要具备以下条件。
 - 该弹性云服务器与目标实例必须处于同一VPC内。
 - 该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。
 - 如果目标实例所属安全组为**默认安全组**，则无需设置安全组规则。
 - 如果目标实例所属安全组**非默认安全组**，请查看安全组规则是否允许该弹性云服务器访问。具体操作请参考[设置安全组规则](#)。
如果安全组规则允许弹性云服务器访问，即可连接实例。
如果安全组规则不允许弹性云服务器访问，则需添加安全组规则。
该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。

2. 使用客户端连接实例。

请参见[如何安装PostgreSQL客户端](#)。

非 SSL 连接

步骤1 登录弹性云服务器或可访问关系型数据库的设备。

步骤2 执行如下命令连接关系型数据库实例。

```
psql --no-readline -U<user>-h<host>-p<port>-d<datastore>-W
```

表 3-9 参数说明

参数	说明
<user>	用户名，即关系型数据库账号（默认管理员账号为root）。
<host>	主机IP，在“实例管理”页面单击实例名称，进入“基本信息”页面。“连接信息”模块的“内网地址”（通过弹性云服务器访问）。
<port>	端口，默认5432，当前端口，即在“实例管理”页面单击实例名称，进入“基本信息”页面，“连接信息”模块的“数据库端口”。
<datastore>	需要连接的数据库名，默认的管理数据库是postgres。

-W是强制用户输入连接密码选项，上述命令执行后，根据提示输入密码。

示例如下：

使用root用户连接postgres数据库实例：

```
psql --no-readline -U root -h 192.168.0.44 -p 5432 -d postgres -W
```

----结束

3.3.3 通过公网连接 PostgreSQL 实例

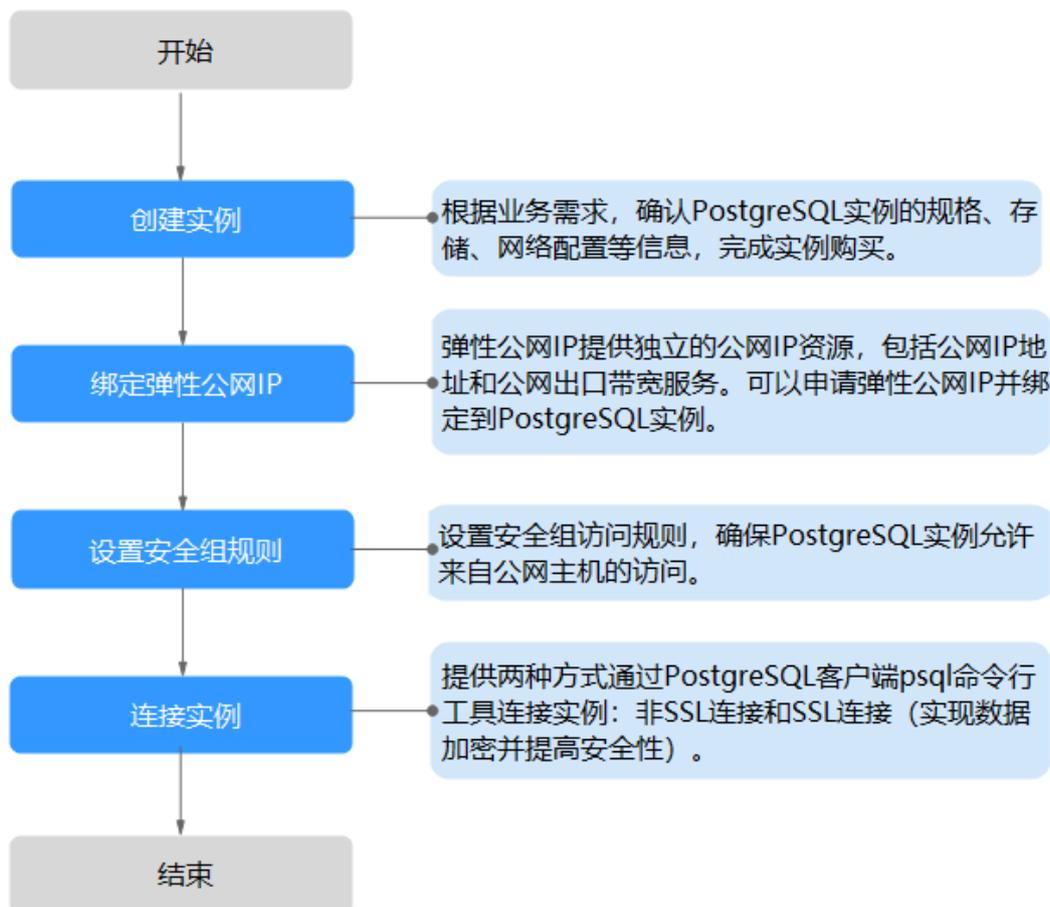
3.3.3.1 公网连接实例流程

首次使用关系型数据库服务，请先了解[PostgreSQL约束与限制](#)。

使用流程

通过公网连接PostgreSQL实例的使用流程介绍如[图3-6](#)所示。

图 3-6 通过公网连接实例



3.3.3.2 绑定弹性公网 IP

操作场景

关系型数据库实例创建成功后，默认未开启公网访问功能（即未绑定弹性公网IP）。关系型数据库服务支持用户绑定弹性公网IP，在公共网络来访问数据库实例，绑定后也可根据需要解绑。

注意事项

- 您需要设置安全组，开通需访问数据库的IP地址和端口，才可以访问数据库实例。在访问数据库前，您需要将访问数据库的IP地址，或者IP段加安全组入方向的访问规则，操作请参见[设置安全组规则](#)。
- 公网访问会降低实例的安全性，请谨慎选择。为了获得更快的传输速率和更高的安全级别，建议您将应用迁移到与您的关系型数据库在同一区域的弹性云服务器上。

绑定弹性公网 IP

步骤1 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤2 在左侧导航栏，单击“弹性公网IP”，单击“绑定弹性公网IP”。

步骤3 在弹出框的弹性公网IP地址列表中，选择目标弹性公网IP，单击“确定”，提交绑定任务。

如果没有可用的弹性公网IP，您可单击“查看弹性公网IP”，获取弹性公网IP。

步骤4 在“弹性公网IP”页面，查看绑定成功的弹性公网IP。

您也可以在“任务中心”页面，查看“实例绑定弹性公网IP”任务的执行进度及结果。

----结束

3.3.3.3 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

通过弹性公网IP连接RDS实例时，需要为RDS所在安全组配置相应的入方向规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建500条安全组规则。
- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的入方向规则。

说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

- 步骤1 在系统首页，单击“网络 > 虚拟私有云”。
 - 步骤2 在左侧导航树选择“访问控制 > 安全组”。
 - 步骤3 在安全组界面，单击操作列的“配置规则”，进入安全组详情界面。
 - 步骤4 在安全组详情界面，单击“添加规则”，弹出添加规则窗口。
 - 步骤5 根据界面提示配置安全组规则。
 - 步骤6 单击“确定”。
- 结束

3.3.3.4 通过 psql 连接实例

通过PostgreSQL客户端连接实例的方式有非SSL连接和SSL连接两种，其中SSL连接通过了加密功能，具有更高的安全性。

前提条件

1. 绑定弹性公网IP并设置安全组规则。
 - a. 对目标实例绑定弹性公网IP。
关于如何绑定弹性公网IP，请参见[绑定弹性公网IP](#)。
 - b. 获取本地设备的IP地址。
 - c. 设置安全组规则。
将1.b中获取的IP地址及目标实例的端口加入安全组允许访问的范围中。
关于如何设置安全组规则，请参见[设置安全组规则](#)章节。
 - d. 使用ping命令连通1.a中绑定的弹性公网IP，确保本地设备可以访问该弹性公网IP。
2. 使用客户端连接实例。
请参见[如何安装PostgreSQL客户端](#)。

非 SSL 连接

- 步骤1 登录弹性云服务器或可访问关系型数据库的设备。
- 步骤2 执行如下命令连接关系型数据库实例。

```
psql --no-readline -U<user>-h<host>-p<port>-d<datastore>-W
```

表 3-10 参数说明

参数	说明
<user>	用户名，即关系型数据库账号（默认管理员账号为root）。
<host>	主机IP，在“实例管理”页面单击实例名称，进入“基本信息”页面。在“弹性公网IP”页签查看弹性公网IP（通过连接了公网的设备访问）。

参数	说明
<code><port></code>	端口，默认5432，当前端口，即在“实例管理”页面单击实例名称，进入“基本信息”页面，“连接信息”模块的“数据库端口”。
<code><datastore></code>	需要连接的数据库名，默认的管理数据库是postgres。

-W是强制用户输入连接密码选项，上述命令执行后，根据提示输入密码。

示例如下：

使用root用户连接postgres数据库实例：

```
psql --no-readline -U root -h 192.168.0.44 -p 5432 -d postgres -W
```

----结束

4 SQL Server 快速入门

4.1 操作指引

本章节提供了如何在管理控制台创建实例、连接实例的具体操作流程。

流程图

图 4-1 操作流程



操作步骤

表 4-1 相关操作及参考手册

相关操作	参考手册
创建RDS实例	步骤一：创建实例
连接RDS实例	步骤二：连接实例

4.2 步骤一：创建实例

操作场景

本节将介绍在关系型数据库服务的管理控制台创建实例的过程。

您可以根据业务需要定制相应计算能力和存储空间的关系型数据库实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击“创建数据库实例”。

步骤5 在“服务选型”页面，填写并选择实例相关信息后，单击“立即创建”。

表 4-2 基本信息

参数	描述
区域	租户当前所在区域，也可在页面左上角切换。 说明 不同区域内的产品内网不互通，且创建后不能更换，请谨慎选择。
实例名称	实例名称长度在4个到64个字符之间，必须以字母开头，可以包含字母、数字、中划线或下划线，不能包含其他特殊字符。
数据库引擎	Microsoft SQL Server。
数据库版本	请参见 数据库引擎和版本 。 不同区域所支持的数据库版本不同，请以实际界面为准。 选用Microsoft SQL Server数据库时，请根据实际业务需求选择合适的数据库引擎版本。建议您选择当前可用的最高版本数据库，因其性能更稳定，安全性更高，使用更可靠。
实例类型+可用区	<ul style="list-style-type: none">主备：一主一备的经典高可用架构。适用于大中型企业的生产数据库，覆盖互联网、物联网、零售电商、物流、游戏等行业应用。备机提高了实例的可靠性，创建主机的过程中，同步创建备机，备机创建成功后，用户不可见。可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。 RDS支持在同一个可用区内或者跨可用区部署数据库主备实例，以提供故障切换能力和高可用性。单机：采用单个数据库节点部署架构，与主流的主备实例相比，它只包含一个节点，但具有高性价比。适用于个人学习、微型网站以及中小企业的开发测试环境。

参数	描述
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">● 高I/O：最大吞吐量150MB/s● 超高I/O：最大吞吐量350MB/s
时区	由于世界各国与地区经度不同，地方时也有所不同，因此会划分为不同的时区。时区仅可在创建实例时选择，实例创建完成后无法修改。
实例字符集	定义数据库或表的排序规则，或应用于字符串表达式时的排序规则强制转换操作。用于设置当前实例的字符集。

表 4-3 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">● 高I/O：最大吞吐量150MB/s● 超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。存储空间支持40GB到4000GB，用户选择容量大小必须为10的整数倍。 创建成功后可进行扩容，具体请参见 磁盘扩容 。

表 4-4 网络

参数	描述
虚拟私有云	关系型数据库实例所在的虚拟专用网络，可以对不同业务进行网络隔离。您需要创建或选择所需的虚拟私有云。如何创建虚拟私有云，请参见《虚拟私有云用户指南》中的“创建虚拟私有云基本信息及默认子网”。 如果没有可选的虚拟私有云，关系型数据库服务默认为您分配资源。 须知 目前RDS实例创建完成后不支持切换虚拟私有云，请谨慎选择所属虚拟私有云。

参数	描述
子网	通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全性。子网在可用区内才会有效，创建关系型数据库实例的子网默认开启DHCP功能，不可关闭。
安全组	安全组限制实例的安全访问规则，加强关系型数据库服务与其他服务间的安全访问。请确保所选取的安全组允许客户端访问数据库实例。 如果没有可选的安全组，关系型数据库服务默认为您分配安全组资源。

表 4-5 数据库配置

参数	描述
管理员账户名	数据库的登录名称默认为rdsuser。
管理员密码	所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*_-+?,特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。 请妥善保管您的密码，因为系统将无法获取您的密码信息。 实例创建成功后，如需重置密码，请参见 重置管理员密码 。
确认密码	必须和主密码相同。
参数模板	数据库参数模板就像是数据库引擎配置值的容器，参数模板中的参数可应用于一个或多个相同类型的数据库实例。对于HA实例创建成功后，主备参数模板相同。实例创建成功后，参数模板可进行修改。 须知 创建数据库实例时，为确保数据库实例正常创建，自定义参数模板中规格相关的参数“max server memory (MB)”不会下发，而是采用系统默认的推荐值。 您可以在实例创建完成之后根据业务需要进行调整。具体请参见 编辑参数模板 。
企业项目	对于已成功关联企业项目的用户，仅需在“企业项目”下拉框中选择目标项目。 如果需要自定义企业项目，请前往项目管理服务进行创建。关于如何创建项目，详见《项目管理用户指南》。

表 4-6 批量创建

参数	描述
创建数量	关系型数据库服务支持批量创建实例，如果您选择创建主备实例，数量选择为1，那么会同步创建一个主实例和一个备实例。 创建多个数据库实例时，名称自动按序增加4位数字后缀。例如输入instance，从instance-0001开始命名；若已有instance-0010，从instance-0011开始命名。

📖 说明

关系型数据库的性能，取决于用户申请关系型数据库时所选择的配置。可供用户选择的硬件配置项为性能规格、存储类型以及存储空间。

步骤6 进行实例规格信息确认。

- 如果需要重新选择实例规格，单击“上一步”，回到上个页面修改关系型数据库实例信息。
- 如果规格确认无误，单击“提交申请”，完成关系型数据库实例的申请。

步骤7 关系型数据库实例创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

- 创建实例过程中，状态显示为“创建中”，创建完成的实例状态显示为“正常”。
- 创建关系型数据库实例时，系统默认开启自动备份策略，后期可修改。实例创建成功后，系统会自动创建一个全量备份。
- 数据库端口默认为1433，实例创建成功后可修改。

具体请参见[修改数据库端口](#)。

----结束

4.3 步骤二：连接实例

4.3.1 实例连接方式介绍

关系型数据库服务提供使用内网、公网的连接方式。

表 4-7 RDS 连接方式

连接方式	IP地址	使用场景	说明
内网连接	内网IP地址	系统默认提供内网IP地址。 当应用部署在弹性云服务器上，且该弹性云服务器与关系型数据库实例处于同一区域，同一VPC时，建议单独使用内网IP连接弹性云服务器与关系型数据库实例。	<ul style="list-style-type: none">• 安全性高，可实现RDS的较好性能。• 推荐使用内网连接。

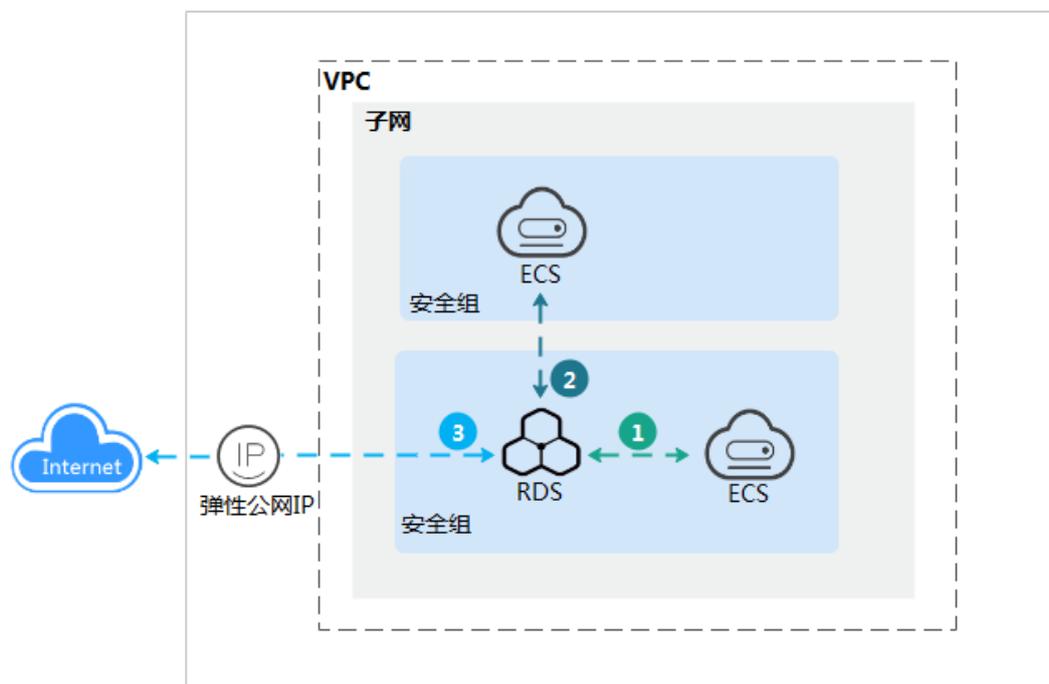
连接方式	IP地址	使用场景	说明
公网连接	弹性公网IP	不能通过内网IP地址访问RDS实例时，使用公网访问，建议单独绑定弹性公网IP连接弹性云服务器（或公网主机）与关系型数据库实例。	<ul style="list-style-type: none"> 安全性低。 为了获得更快的传输速率和更高的安全性，建议您将应用迁移到与您的RDS实例在同一VPC内，使用内网连接。

说明

- VPC：虚拟私有云（Virtual Private Cloud，简称VPC）。
- ECS：弹性云服务器（Elastic Cloud Server，简称ECS）。
- 若弹性云服务器和关系型数据库实例处于同一个VPC内，则无需申请外网地址。

其中，通过内网和公网的连接方式如图4-2所示。

图 4-2 实例内网和公网连接



- 1 通过内网连接RDS实例（ECS与RDS在相同安全组）
- 2 通过内网连接RDS实例（ECS与RDS在不同安全组）
- 3 通过公网连接RDS实例

4.3.2 通过内网连接 SQL Server 实例

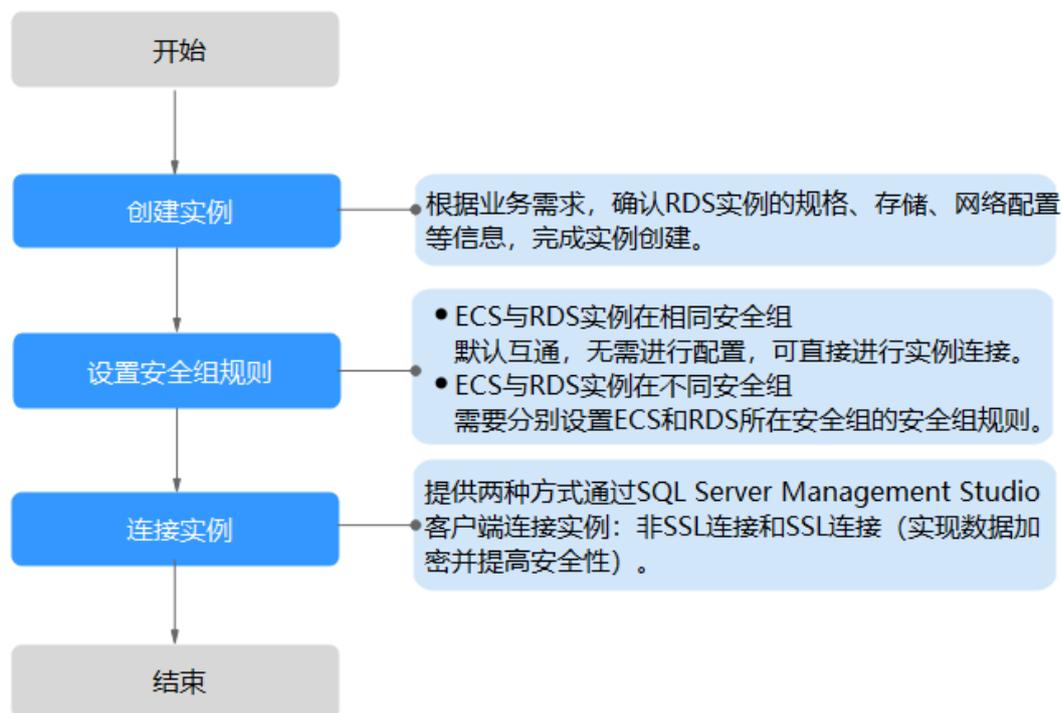
4.3.2.1 内网连接实例流程

若您第一次使用关系型数据库，请先了解[Microsoft SQL Server约束与限制](#)。

使用流程

通过内网连接SQL Server实例的使用流程介绍如[图4-3](#)所示。

图 4-3 通过内网连接实例



4.3.2.2 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

内网连接RDS实例时，设置安全组分为以下两种情况：

- ECS与RDS实例在相同安全组时，默认ECS与RDS实例互通，无需设置安全组规则，执行[通过内网连接SQL Server实例](#)。
- ECS与RDS实例在不同安全组时，需要为RDS和ECS分别设置安全组规则。
 - 设置RDS安全组规则：为RDS所在安全组配置相应的入方向规则。

- 设置ECS安全组规则：安全组默认规则为出方向上数据报文全部放行，此时，无需对ECS配置安全组规则。当在ECS所在安全组为非默认安全组且出方向规则**非全放通**时，需要为ECS所在安全组配置相应的**出方向**规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建100个安全组。
- 默认情况下，一个安全组最多只允许拥有50条安全组规则。
- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的**入方向**规则。

说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

- 步骤1** 登录管理控制台。
 - 步骤2** 在系统首页，单击“网络 > 虚拟私有云”。
 - 步骤3** 在左侧导航树，单击“安全组”。
 - 步骤4** 在安全组界面，单击操作列的“配置规则”，进入安全组详情界面。
 - 步骤5** 在安全组详情界面，单击“添加规则”，弹出添加规则窗口。
 - 步骤6** 根据界面提示配置安全组规则。
 - 步骤7** 单击“确定”。
- 结束

4.3.2.3 通过内网连接 SQL Server 实例

提供两种连接方式通过SQL Server客户端连接实例：非SSL连接和SSL连接。其中，SSL连接实现了数据加密功能，具有更高的安全性。

准备工作

1. 准备弹性云服务器

通过内网连接关系型数据库实例，您需要创建一台弹性云服务器。

创建并连接弹性云服务器，请参见[如何创建和连接ECS](#)。

- 该弹性云服务器与目标实例必须处于同一VPC内。
- 该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。
 - 如果目标实例所属安全组为**默认安全组**，则无需设置安全组规则。

- 如果目标实例所属安全组**非默认安全组**，请查看安全组规则是否允许该弹性云服务器访问。
 - 1) 登录管理控制台。
 - 2) 单击管理控制台左上角的📍，选择区域和项目。
 - 3) 选择“数据库 > 关系型数据库”。进入关系型数据库信息页面。
 - 4) 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。
 - 5) 在“连接信息”模块的“安全组”处，单击安全组名称，进入安全组页面，查看安全组规则。如果安全组规则允许弹性云服务器访问，即可连接实例。
如果安全组规则不允许弹性云服务器访问，则需添加安全组规则。
具体操作请参考[设置安全组规则](#)。

2. 安装SQL Server客户端

在1中的弹性云服务器或可访问关系型数据库实例的设备上，安装Microsoft SQL Server客户端。

非 SSL 连接

- 步骤1** 登录弹性云服务器或可访问关系型数据库实例的设备。
- 步骤2** 启动SQL Server Management Studio客户端。
- 步骤3** 选择“连接 > 数据库引擎”，在“连接到服务器”弹出框中输入登录信息。

图 4-4 连接到服务器



- “服务器名称”是目标实例的主机IP和数据库端口（IP和数据库端口之间请使用英文半角逗号）。例如：x.x.x.x,8080
 - 主机IP为“基本信息”页签中，“连接信息”模块的“内网地址”。
 - 端口为“基本信息”页签中，“连接信息”模块的“数据库端口”。

- “身份验证”是认证方式，选择“SQL Server身份验证”。
- “登录名”即待访问的关系型数据库账号，默认管理员账号为rdsuser。
- “密码”即待访问的数据库账号对应的密码。

步骤4 单击“连接”，连接实例。

📖 说明

若连接失败，请确保各项准备工作正确配置后，重新尝试连接。

----结束

SSL 连接

步骤1 下载并上传SSL根证书。

1. 在“实例管理”页面，单击实例名称进入“基本信息”页面，单击“数据库信息”模块“SSL”处的 ，下载根证书或捆绑包。
2. 将根证书上传至需要连接SQL Server实例的弹性云服务器，或保存到可访问关系型数据库实例的设备。
3. 将根证书导入弹性云服务器Windows操作系统，请参见[将根证书导入Windows/Linux操作系统](#)。

📖 说明

- 请在原有根证书到期前及时更换正规机构颁发的证书，以提高系统安全性。
- 对于SQL Server，绑定公网IP后，需重启实例才能使SSL连接生效。

步骤2 启动SQL Server Management Studio客户端。

步骤3 选择“连接 > 数据库引擎”，在“连接到服务器”弹出框中填选登录信息。

图 4-5 连接到服务器



- “服务器名称”是目标实例的主机IP和数据库端口（IP和数据库端口之间请使用英文半角逗号）。例如：x.x.x.x,8080

- 主机IP为“基本信息”页签中，“连接信息”模块的“内网地址”。
- 端口为“基本信息”页签中，“连接信息”模块的“数据库端口”。
- “身份验证”是认证方式，选择“SQL Server身份验证”。
- “登录名”即待访问的关系型数据库账号，默认管理员账号为rdsuser。
- “密码”即待访问的数据库账号对应的密码。

步骤4 在“连接属性”页签，填选相关信息，并勾选“加密连接”，启用SSL加密（系统默认不勾选“加密连接”，即不启用，需手动启用）。

图 4-6 连接属性



步骤5 单击“连接”，连接实例。

说明

若连接失败，请确保各项准备工作正确配置后，重新尝试连接。

----结束

4.3.3 通过公网连接 SQL Server 实例

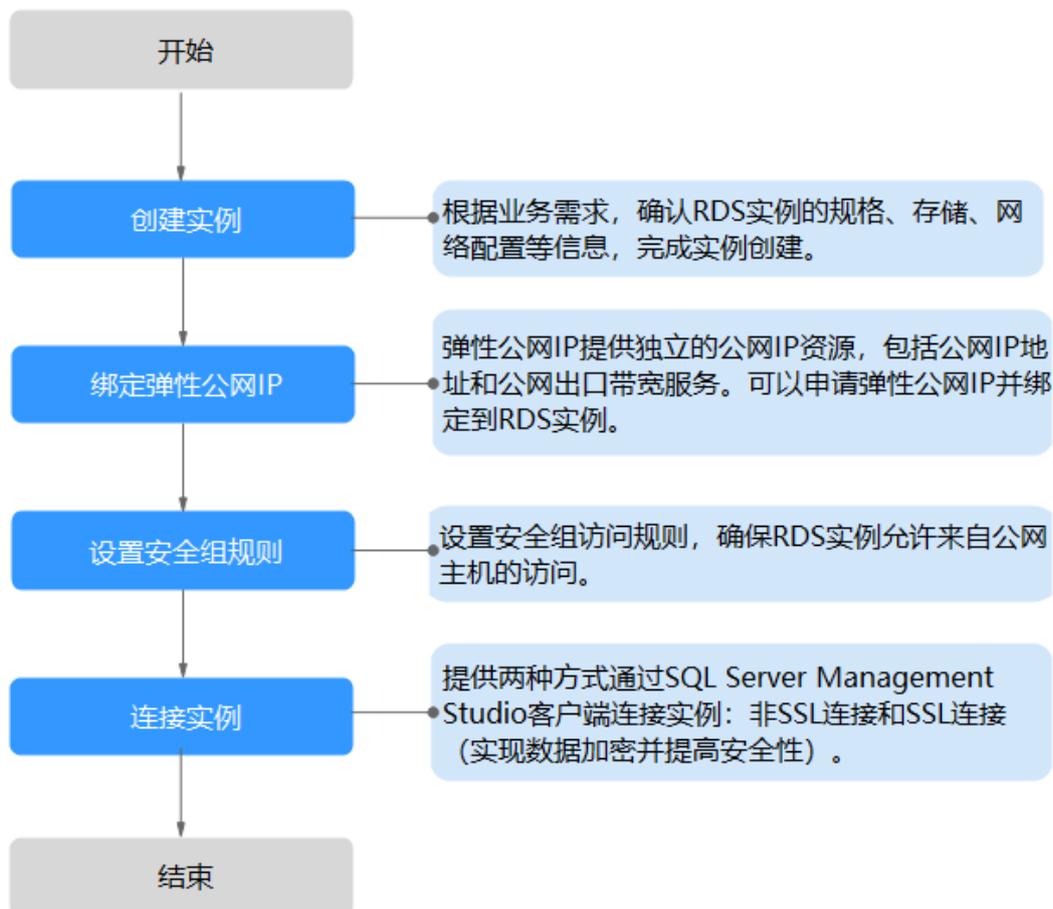
4.3.3.1 公网连接实例流程

若您第一次使用关系型数据库，请先了解RDS的[Microsoft SQL Server约束与限制](#)。

使用流程

通过公网连接SQL Server实例的使用流程介绍如[图4-7](#)所示。

图 4-7 通过公网连接实例



4.3.3.2 绑定弹性公网 IP

操作场景

关系型数据库实例创建成功后，默认未开启公网访问功能（即未绑定弹性公网IP）。关系型数据库服务支持用户绑定弹性公网IP，在公共网络来访问数据库实例，绑定后也可根据需要解绑。

注意事项

- 您需要设置安全组，开通需访问数据库的IP地址和端口，才可以访问数据库实例。在访问数据库前，您需要将访问数据库的IP地址，或者IP段加安全组入方向的访问规则，操作请参见[设置安全组规则](#)。

绑定弹性公网 IP

- 步骤1** 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。
- 步骤2** 在左侧导航栏，单击“弹性公网IP”，单击“绑定弹性公网IP”。
- 步骤3** 在弹出框的弹性公网IP地址列表中，选择目标弹性公网IP，单击“确定”，提交绑定任务。

如果没有可用的弹性公网IP，您可单击“查看弹性公网IP”，获取弹性公网IP。

步骤4 在“弹性公网IP”页面，查看绑定成功的弹性公网IP。

您还可以在“任务中心”页面，查看“实例绑定弹性公网IP”任务的执行进度及结果。

----结束

4.3.3.3 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

通过弹性公网IP连接RDS实例时，需要为RDS所在安全组配置相应的入方向规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建500条安全组规则。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的入方向规则。

说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

步骤1 登录管理控制台。

步骤2 在系统首页，单击“网络 > 虚拟私有云”。

步骤3 在左侧导航树选择“访问控制 > 安全组”。

步骤4 在安全组界面，单击操作列的“配置规则”，进入安全组详情界面。

步骤5 在安全组详情界面，单击“添加规则”，弹出添加规则窗口。

步骤6 根据界面提示配置安全组规则。

步骤7 单击“确定”。

----结束

4.3.3.4 通过公网连接 SQL Server 实例

提供两种连接方式通过SQL Server客户端连接实例：非SSL连接和SSL连接。其中，SSL连接实现了数据加密功能，具有更高的安全性。

准备工作

1. 安装SQL Server客户端
请参见[如何安装SQL Server Management Studio](#)。
2. 绑定弹性公网IP并设置安全组规则
 - a. 对目标实例绑定弹性公网IP。
关于如何绑定弹性公网IP，请参见[绑定弹性公网IP](#)。
 - b. 获取本地设备的IP地址。
 - c. 设置安全组规则。
将2.b中获取的IP地址及目标实例的端口加入安全组允许访问的范围中。
关于如何设置安全组规则，请参见[设置安全组规则](#)章节。
 - d. 使用ping命令连通2.a中绑定的弹性公网IP地址，确保本地设备可以访问该弹性公网IP地址。

非 SSL 连接

步骤1 启动SQL Server Management Studio客户端。

步骤2 选择“连接 > 数据库引擎”，在“连接到服务器”弹出框中填选登录信息。

图 4-8 连接到服务器



- “服务器名称”是目标实例的主机IP和数据库端口（IP和数据库端口之间请使用英文半角逗号）。例如：x.x.x.x,8080
 - 主机IP为已绑定的弹性公网IP地址。
 - 端口为“基本信息”页签中，“连接信息”模块的“数据库端口”。

- “身份验证”是认证方式，选择“SQL Server身份验证”。
- “登录名”即待访问的关系型数据库账号，默认管理员账号为rdsuser。
- “密码”即待访问的数据库账号对应的密码。

步骤3 单击“连接”，连接实例。

📖 说明

若连接失败，请确保各项准备工作正确配置后，重新尝试连接。

---结束

SSL 连接

步骤1 下载并上传SSL根证书。

1. 在“实例管理”页面，单击实例名称进入“基本信息”页面，单击“数据库信息”模块“SSL”处的 ，下载根证书或捆绑包。
2. 将根证书上传至需要连接SQL Server实例的弹性云服务器。
3. 将根证书导入弹性云服务器Windows操作系统，请参见[将根证书导入Windows/Linux操作系统](#)。

📖 说明

- 请在原有根证书到期前及时更换正规机构颁发的证书，以提高系统安全性。
- 对于SQL Server，绑定公网IP后，需重启实例才能使SSL连接生效。

步骤2 启动SQL Server Management Studio客户端。

步骤3 选择“连接 > 数据库引擎”，在“连接到服务器”弹出框中填选登录信息。

图 4-9 连接到服务器



- “服务器名称”是目标实例的主机IP和数据库端口（IP和数据库端口之间请使用英文半角逗号）。例如：x.x.x.x,8080

- 主机IP为已绑定的弹性公网IP地址。
- 端口为“基本信息”页签中，“连接信息”模块的“数据库端口”。
- “身份验证”是认证方式，选择“SQL Server身份验证”。
- “登录名”即待访问的关系型数据库账号，默认管理员账号为rdsuser。
- “密码”即待访问的数据库账号对应的密码。

步骤4 在“连接属性”页签，填选相关信息，并勾选“加密连接”，启用SSL加密（系统默认不勾选“加密连接”，即不启用，需手动启用）。

图 4-10 连接属性



步骤5 单击“连接”，连接实例。

📖 说明

若连接失败，请确保各项准备工作正确配置后，重新尝试连接。

----结束

5 MySQL 用户指南

5.1 数据迁移

5.1.1 使用 mysqldump 迁移 MySQL 数据

迁移准备

关系型数据库服务支持开启公网访问功能，通过弹性公网IP进行访问。您也可通过弹性云服务器的内网访问关系型数据库。

1. 准备弹性云服务器或可通过公网访问关系型数据库。
 - 通过弹性云服务器连接关系型数据库实例，需要创建一台弹性云服务器。
 - 通过公网地址连接关系型数据库实例，需具备以下条件。
 - i. 先对关系型数据库实例绑定公网地址，如何绑定公网地址，请参见[绑定弹性公网IP](#)。
 - ii. 保证本地设备可以访问关系型数据库实例绑定的公网地址。
2. 在准备的弹性云服务器或可访问关系型数据库的设备上，安装MySQL客户端。

说明

该弹性云服务器或可访问关系型数据库的设备需要安装和RDS MySQL数据库服务端相同版本的数据库客户端，MySQL数据库或客户端会自带mysqldump和mysql工具。

数据迁移到云数据库RDS后可能要面对更改IP的问题，为减少客户业务更改，降低迁移难度，支持更改内网IP，具体请参见[配置和修改内网地址](#)。

导出数据

要将源数据库迁移到关系型数据库，需要先对其进行导出。

须知

- 相应导出工具需要与数据库引擎版本匹配。
- 数据库迁移为离线迁移，您需要停止使用源数据库的应用程序。

- 步骤1** 登录已准备的弹性云服务器，或可访问关系型数据库的设备。
- 步骤2** 使用mysqldump将元数据导出至SQL文件。

须知

MySQL数据库是关系型数据库服务管理所必须的数据库，导出元数据时，禁止指定**--all-database**参数，否则会造成数据库故障。

```
mysqldump --databases <DB_NAME> --single-transaction --order-by-primary  
--hex-lob --no-data --routines --events --set-gtid-purged=OFF -u <DB_USER>  
-p -h <DB_ADDRESS> -P <DB_PORT> |sed -e 's/DEFINER[ ]*=[ ]*[^\]*\*/' -e 's/  
DEFINER[ ]*=.*/FUNCTION/FUNCTION/' -e 's/DEFINER[ ]*=.*/PROCEDURE/  
PROCEDURE/' -e 's/DEFINER[ ]*=.*/TRIGGER/TRIGGER/' -e 's/  
DEFINER[ ]*=.*/EVENT/EVENT/' > <BACKUP_FILE>
```

- DB_NAME为要迁移的数据库名称。
- DB_USER为数据库用户。
- DB_ADDRESS为数据库地址。
- DB_PORT为数据库端口。
- BACKUP_FILE为导出生成的文件名称。

根据命令提示输入数据库密码。

示例如下：

```
mysqldump --databases rdsdb --single-transaction --order-by-primary --hex-  
lob --no-data --routines --events --set-gtid-purged=OFF -u root -p -h  
192.168.151.18 -P 3306 |sed -e 's/DEFINER[ ]*=[ ]*[^\]*\*/' -e 's/  
DEFINER[ ]*=.*/FUNCTION/FUNCTION/' -e 's/DEFINER[ ]*=.*/PROCEDURE/  
PROCEDURE/' -e 's/DEFINER[ ]*=.*/TRIGGER/TRIGGER/' -e 's/  
DEFINER[ ]*=.*/EVENT/EVENT/' > dump-defs.sql
```

Enter password:

说明

若使用的mysqldump低于5.6版本，需要去掉“--set-gtid-purged=OFF”。

命令执行完会生成“dump-defs.sql”文件，如下：

```
[rds@localhost ~]$ ll dump-defs.sql  
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 dump-defs.sql
```

- 步骤3** 使用mysqldump将数据导出至SQL文件。

须知

MySQL数据库是关系型数据库服务管理所必须的数据库，导出元数据时，禁止指定**--all-database**参数，否则会造成数据库故障。

```
mysqldump --databases <DB_NAME> --single-transaction --hex-blob --set-gtid-purged=OFF --no-create-info --skip-triggers -u <DB_USER> -p -h <DB_ADDRESS> -P <DB_PORT> -r <BACKUP_FILE>
```

以上命令的参数说明如[步骤2](#)所示。

根据命令提示输入数据库密码。

示例如下：

```
mysqldump --databases rdsdb --single-transaction --hex-blob --set-gtid-purged=OFF --no-create-info --skip-triggers -u root -p -h 192.168.151.18 -P 8635 -r dump-data.sql
```

📖 说明

若使用的mysqldump低于5.6版本，需要去掉“--set-gtid-purged=OFF”。

命令执行完会生成“dump-data.sql”文件，如下：

```
[rds@localhost ~]$ ll dump-data.sql  
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 dump-data.sql
```

----结束

导入数据

通过弹性云服务器或可访问关系型数据库的设备，用相应客户端连接关系型数据库实例，将导出的SQL文件导入到关系型数据库。

须知

如果源数据库中包含触发器、存储过程、函数或事件调用，则需确保导入前设置目标数据库参数log_bin_trust_function_creators=ON。

步骤1 登录已准备的弹性云服务器，或可访问关系型数据库的设备。

步骤2 导入元数据到关系型数据库。

```
# mysql -f -h <RDS_ADDRESS> -P <DB_PORT> -u root -p <<BACKUP_DIR>/  
dump-defs.sql
```

- RDS_ADDRESS为关系型数据库实例的IP地址。
- DB_PORT为当前数据库实例的端口。
- BACKUP_DIR为“dump-defs.sql”所在目录。

示例如下：

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p < dump-defs.sql
```

Enter password:

📖 说明

如果是将单表导出的数据（即导出的SQL是某一张表的SQL语句，不包含数据库）再进行导入，建议指定待导入的数据库（mydb），否则可能会报错“No database selected”。示例：

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p mydb < dump-defs.sql
Enter password:
```

步骤3 导入数据到关系型数据库。

```
# mysql -f -h <RDS_ADDRESS> -P <DB_PORT> -u root -p <<BACKUP_DIR>/
dump-data.sql
```

- RDS_ADDRESS为关系型数据库实例的IP地址。
- DB_PORT为当前数据库实例的端口。
- BACKUP_DIR为“dump-data.sql”所在目录。

示例如下：

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p < dump-data.sql
Enter password:
```

📖 说明

如果是将单表导出的数据（即导出的SQL是某一张表的SQL语句，不包含数据库）再进行导入，建议指定待导入的数据库（mydb），否则可能会报错“No database selected”。示例：

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p mydb < dump-defs.sql
Enter password:
```

步骤4 查看迁移结果。

```
mysql> show databases;
```

示例中，名为rdsdb的数据库已经被导入了：

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

----结束

5.2 参数调优

5.2.1 MySQL 参数调优建议

数据库参数是数据库系统运行的关键配置信息，设置不合适的参数值可能会影响业务。本文列举了一些重要参数说明，更多参数详细说明，请参见[MySQL官网](#)。

通过控制台界面修改MySQL参数值，请参见[编辑参数模板](#)。

修改敏感参数

若干参数相关说明如下：

- “lower_case_table_names”

云数据库默认值：“1”。

作用：该参数表示创建数据库及表时，表存储是否大小写敏感。默认值“1”，表示创建数据库及表时，默认小写，不区分大小写。

📖 说明

MySQL 8.0版本不支持该参数。

影响：修改该参数可能会导致主从复制异常，请谨慎修改。如果必须要修改，请根据以下场景设置数据库参数：

- 参数值从1变为0的设置顺序：先修改和重启只读库，后修改和重启主库。
- 参数值从0变为1的设置顺序：先修改和重启主库，在主库执行**SELECT @@GLOBAL.GTID_EXECUTED**。然后在只读库执行**SELECT @@GLOBAL.GTID_EXECUTED**，直到结果集合大于或者等于主库的**SELECT @@GLOBAL.GTID_EXECUTED**的结果集合，再修改和重启只读库。

- “innodb_flush_log_at_trx_commit”

云数据库默认值：“1”。

作用：该参数控制提交操作在严格遵守ACID合规性和高性能之间的平衡。设置为默认值“1”，是为了保证完整的ACID，每次提交事务时，把事务日志从缓存区写到日志文件中，并刷新日志文件的数据到磁盘上；当设为“0”时，每秒把事务日志缓存区的数据写入日志文件，并刷新到磁盘；如果设为“2”，每次提交事务都会把事务日志从缓存区写入日志文件，每隔一秒左右会刷新到磁盘。

影响：参数设置为非默认值“1”时，降低了数据安全性，在系统崩溃的情况下，可能导致数据丢失。

- “sync_binlog”

云数据库默认值：“1”。

作用：该参数控制MySQL服务器将二进制日志同步到磁盘的频率。设置为默认值“1”，表示MySQL每次事务提交，binlog同步写入磁盘，是最安全的设置；设置为“0”时，表示MySQL不控制binlog的刷新，由文件系统自己控制其缓存的刷新。此时的性能最好，但风险最大，因为一旦断电或操作系统崩溃，在“binlog_cache”中的所有binlog信息都会被丢失。

影响：参数设置为非默认值“1”时，降低了数据安全性，在系统崩溃的情况下，可能导致binlog丢失。

- “innodb_large_prefix”

云数据库默认值：“OFF”。

作用：InnoDB表允许单列索引的最大长度。

📖 说明

仅MySQL 5.6版本支持该参数。

影响：在DDL执行时修改该参数，有可能会主从复制异常，请谨慎修改。如果必须要修改，请根据以下场景设置数据库参数：

- 参数值从OFF变为ON的设置顺序：先修改只读库，后修改主库。
- 参数值从ON变为OFF的设置顺序：先修改主库，后修改只读库。

修改性能参数

若干参数相关说明如下：

- “innodb_spin_wait_delay” 和 “query_alloc_block_size” 依赖于实例的规格，设置过大时，可能会影响数据库的使用。
- “key_buffer_size” 参数值设置较小（小于4096），参数值将修改失败。
- “max_connections” 参数值设置较小，将影响数据库访问。
- “innodb_buffer_pool_size”、“max_connections” 和 “back_log” 参数依赖于实例的规格，实例规格不同对应其默认值也不同。因此，这些参数在用户未设置前显示为 “default”。
- “innodb_io_capacity_max”、“innodb_io_capacity” 参数依赖于磁盘类型，用户未设置前显示为 “default”。

5.3 MySQL 内核功能

5.3.1 内核版本说明

本章节介绍云数据库 RDS for MySQL的内核版本更新说明。

RDS for MySQL 8.0

表 5-1 云数据库 RDS for MySQL 8.0 版本说明

日期	特性描述
2022-12-09	<ul style="list-style-type: none">• 新特性及性能优化<ul style="list-style-type: none">- KILL会话支持per thread连接方式。- Performance Schema使用增加内存限制。- 优化特定场景下全量SQL的性能。- 优化特定场景下internal_tmp_mem_storage_engine为memory引擎的性能问题。- 升级编译器到GCC 10.3。• 修复问题<ul style="list-style-type: none">- 修复临时文件写入可能报错的问题。- 修复CTE查询返回结果不符合预期的问题。• 安全加固<ul style="list-style-type: none">- 解决安全漏洞：CVE-2021-2417、CVE-2021-2339、CVE-2021-2425、CVE-2021-2426、CVE-2021-2427、CVE-2021-2424、CVE-2021-2383、CVE-2021-2384、CVE-2021-2410。

日期	特性描述
2022-09-01	<ul style="list-style-type: none"> ● 新特性及性能优化 <ul style="list-style-type: none"> - 内核版本升级到8.0.25。 - 支持SQL限流。 - 升级编译器到GCC 9.3。 ● 修复问题 <ul style="list-style-type: none"> - 修复单个超大Binlog可能导致的复制中断问题。 - 修复社区innodb_row_lock_current_waits统计不准确的问题。 - 修复使用blob字段可能导致的异常重启问题。 ● 安全加固 <ul style="list-style-type: none"> - 解决安全漏洞：CVE-2021-2307、CVE-2021-2180、CVE-2021-2194。
2021-11-07	<ul style="list-style-type: none"> ● 新特性及性能优化 <ul style="list-style-type: none"> - 线程池静态链接，提高性能。 - 开启PGO（Profile-Guided Optimization）编译优化。 - 优化MySQL HASH算法。 - 支持数据库添加备注功能。 - 系统库防止被DDL修改。 - 添加参数innodb_total_tablespace用于统计innodb表空间数量。 - 在i_s中提供innodb锁视图。 - 升级openssl, jemalloc, curl开源组件。 ● 修复问题 <ul style="list-style-type: none"> - 修复XA事务在binlog rotate后可能异常重启的问题。 - 修复全量SQL未记录预编译SQL操作类型的问题。 - 修复flush privieges操作的executed time统计错误的问题。 - 修复审计日志错误写入其他文件的问题。
2021-07-13	<ul style="list-style-type: none"> ● 修复XA事务在数据库异常重启后可能发生丢失的问题。 ● 优化自适应HASH分割算法。 ● 内核版本升级到8.0.21。 ● 安全加固 <ul style="list-style-type: none"> 解决安全漏洞：CVE-2020-14697、CVE-2020-14680、CVE-2020-14678、CVE-2020-14663、CVE-2021-2020、CVE-2020-14619、CVE-2020-14591、CVE-2020-14576、CVE-2020-14539。
2021-04-26	<ul style="list-style-type: none"> ● SQL限流特性性能优化。 ● 全量SQL采集性能优化。

日期	特性描述
2021-03-31	<ul style="list-style-type: none"> 性能优化 升级编译器到GCC 9。
2021-03-01	<ul style="list-style-type: none"> 性能优化 优化慢日志额外信息的采集效率。 修复以下问题 修复XA事务回滚可能会导致备机复制中断问题。
2021-02-06	<ul style="list-style-type: none"> 修复问题 修复微秒计时器中gettimeofday多线程导致计时错误问题。
2020-12-21	<ul style="list-style-type: none"> 支持在错误日志中记录详细的连接断开信息。 支持index hint功能。
2020-11-03	<ul style="list-style-type: none"> 显示大事务执行时间和等待时间。 为管理用户使用独立的连接控制。 支持SQL限流，业务高峰期可以限制特定SQL的执行频率。 优化内核性能。

RDS for MySQL 5.7

表 5-2 云数据库 RDS for MySQL 5.7 版本说明

日期	特性描述
2022-12-09	<ul style="list-style-type: none"> 新特性及性能优化 <ul style="list-style-type: none"> 回合升级到5.7.38版本。 升级编译器到GCC 10.3。 KILL会话支持per thread连接方式。 慢日志的阈值可考虑等锁耗时。 ALT功能安全加固。 修复问题 <ul style="list-style-type: none"> 增强分布式XA事务主库崩溃恢复安全。 修复开启proxy可能导致的异常重启。 修复插件可能内存申请失败导致的异常重启。 安全加固 <ul style="list-style-type: none"> 解决安全漏洞：CVE-2022-21454、CVE-2022-21417、CVE-2022-21427、CVE-2022-21451、CVE-2022-21444、CVE-2022-21460。

日期	特性描述
2022-09-01	<ul style="list-style-type: none">● 新特性及性能优化<ul style="list-style-type: none">- 内核版本升级到5.7.37。- 升级编译器到GCC 9.3。- 升级openssl、curl开源组件。● 修复问题<ul style="list-style-type: none">- 修复单个超大Binlog可能导致的复制中断问题。- 修复slave并发回放grant可能导致的异常重启的问题。- 修复隐藏自增键可能导致的复制中断问题。- 修复虚拟字段表在ROLLBACK时可能异常重启的问题。- 修复加密表在recovery异常重启问题。- 修复Seconds Behind Master特定场景不准确问题。● 安全加固<ul style="list-style-type: none">- 解决安全漏洞：CVE-2022-21367、CVE-2022-21304、CVE-2022-21344。
2022-04-26	<ul style="list-style-type: none">● 新特性及性能优化<ul style="list-style-type: none">- 支持隐藏自增键特性。● 修复问题<ul style="list-style-type: none">- 修复XA事务重复提交可能导致复制异常问题。- 修复社区innodb_row_lock_current_waits统计不准确的问题。
2022-02-26	<ul style="list-style-type: none">● 新特性及性能优化<ul style="list-style-type: none">- 全量SQL放开单条记录长度限制。- RDS for MySQL透明应用连续性(ALT)特性第一期。● 修复问题<ul style="list-style-type: none">- 修复线程池极端场景的内存问题。- 修复XA事务备机回放概率性卡住。

日期	特性描述
2021-11-07	<ul style="list-style-type: none"> ● 新特性及性能优化 <ul style="list-style-type: none"> - 线程池静态链接，提高性能。 - 开启PGO (Profile-Guided Optimization) 编译优化。 - 优化MySQL HASH算法。 - 支持数据库添加备注功能。 - 系统库防止被DDL修改。 - 在errlog中提供IO延迟信息。 - 支持minidump。 - 内核版本升级到5.7.33。 - 升级openssl, jemalloc, curl开源组件。 ● 修复问题 <ul style="list-style-type: none"> - 修复由于回放保序导致备机小概率复制中断的问题。 - 修复XA事务在binlog rotate后可能异常重启的问题。 - 修复全量SQL可能存在未记录预编译SQL操作类型的问题。 - 修复flush privieges操作的executed time统计错误的问题。 - 修复审计日志错误写入其他文件的问题。 ● 安全加固 解决安全漏洞：CVE-2021-2011、CVE-2021-2178、CVE-2021-2202。
2021-07-13	<ul style="list-style-type: none"> ● 修复XA事务在数据库异常重启后可能发生丢失的问题。 ● 优化自适应HASH分割算法。 ● 内核版本升级到5.7.32。
2021-04-26	<ul style="list-style-type: none"> ● 新特性 支持在代理情况下显示真实的客户端地址。 ● 修复问题 全量SQL采集性能优化。 修复revoke权限可能导致主备权限不一致的问题。 回合8.0中对Instant add column的优化。
2021-03-31	<ul style="list-style-type: none"> ● 性能优化 <ul style="list-style-type: none"> - 优化慢日志额外信息的采集效率。 - 升级编译器至GCC 9。 ● 修复问题 修复XA事务回滚可能会导致备机复制中断问题。
2021-03-01	减少fil_sys互斥锁之间的冲突频率。

日期	特性描述
2021-02-06	<ul style="list-style-type: none"> ● 新特性 <ul style="list-style-type: none"> - 内存申请编译器优化。 - 添加utf8优化对非ARM平台的支持。 ● 修复问题 <ul style="list-style-type: none"> 修复微秒计时器中gettimeofday多线程导致计时错误问题。
2020-12-21	<ul style="list-style-type: none"> ● 内核版本升级到5.7.31。 ● 支持SQL限流，业务高峰期可以限制特定SQL的执行频率。
2020-11-03	<ul style="list-style-type: none"> ● 优化内核性能。 ● 支持回收站。 ● 增强本地盘清理数据可靠性。

RDS for MySQL 5.6

表 5-3 云数据库 RDS for MySQL 5.6 版本说明

日期	特性描述
2022-12-09	<ul style="list-style-type: none"> ● 新特性及性能优化 <ul style="list-style-type: none"> - KILL会话支持per thread连接方式。 ● 修复问题 <ul style="list-style-type: none"> - 修复开启proxy可能导致的异常重启。 - 修复main.proxy_connect缓冲区溢出。 - 修复插件可能内存申请失败导致的异常重启。
2022-09-01	修复社区innodb_row_lock_current_waits统计不准确的问题。
2021-11-07	<ul style="list-style-type: none"> ● 新特性 <ul style="list-style-type: none"> - 支持数据库添加备注功能。 - 系统库防止被DDL修改。 - 升级openssl, jemalloc开源组件。 ● 修复问题 <ul style="list-style-type: none"> - 修复修改密码可能会导致同步断开的问题。 - 修复审计日志错误写入其他文件的问题。
2021-07-13	<ul style="list-style-type: none"> ● 修复由于回放保序导致备机小概率复制中断的问题。 ● 内核版本升级到5.6.51。 ● 安全加固 <ul style="list-style-type: none"> 社区MySQL 5.6版本补丁不再发布，该版本的漏洞补丁服务也不再发布。

日期	特性描述
2021-04-26	<ul style="list-style-type: none">● 新特性 支持在代理情况下显示真实的客户端地址。● 修复问题 修复执行select 1 for update报语法错误问题。 全量SQL采集性能优化。
2021-03-31	重构SQL限流功能，提高易用性。
2021-02-06	内核版本升级到5.6.50。
2020-12-23	支持SQL限流，业务高峰期可以限制特定SQL的执行频率。
2020-11-03	内核版本升级到5.6.49。
2020-10-09	<ul style="list-style-type: none">● 支持本地盘磁盘日志切换。● 支持在错误日志中记录用户的操作历史。

5.4 实例生命周期

5.4.1 创建与原实例相同配置的数据库实例

操作场景

关系型数据库服务支持用户快速创建相同配置的数据库实例。

📖 说明

- 创建相同配置实例没有次数限制。
- 只读实例不能执行创建相同配置的操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 创建相同配置”。

步骤5 在“创建相同配置”页面，各项配置默认和原实例信息相同，您可根据业务需要进行修改，填选完毕后，单击“立即创建”。

MySQL实例配置相关信息，请参见[步骤一：创建实例](#)。

步骤6 确认实例信息。

步骤7 稍后刷新实例列表，查看创建结果。如果实例状态为“正常”，说明实例创建成功。

创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

----结束

5.4.2 重启实例

操作场景

通常出于维护目的，您可能需要重启数据库实例。例如：对于某些运行参数修改，需要重启单个实例使之生效。您可通过控制台对主实例和只读实例执行重启操作。

约束限制

- 如果数据库实例处于“异常”状态，可能会重启失败。
- 重启数据库实例会重新启动数据库引擎服务。重启数据库实例将导致短暂中断，在此期间，数据库实例状态将显示为“重启中”。
- 重启过程中，实例将不可用。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主实例，或者单击  展开只读实例，单击“更多 > 重启实例”。

您也可以在“实例管理”页面单击目标实例名称，进入“基本信息”页面。在页面右上角，单击“重启实例”。

重启实例时，如果是主备实例，对应的备实例也会被同步重启。

步骤5 在“重启实例”弹框，单击“是”重启实例。

步骤6 稍后刷新实例列表，查看重启结果。如果实例状态为“正常”，说明实例重启成功。

----结束

5.4.3 自定义列表项

操作场景

您可以根据自身业务需要，自定义设置实例信息列表项。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击，选择自定义列表项。

- 系统默认展示以下列表项：实例名称/ID、实例类型、数据库引擎版本、运行状态、内网地址、操作。
默认实例信息列表项不可更改。
- 单project情况下，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。
- 多project情况下，需要同时开通项目管理服务权限，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。

----结束

5.4.4 导出实例

操作场景

您可以导出所有实例，或根据一定条件筛选出来的目标实例，查看并分析实例信息。

导出所有实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例列表右上角，默认导出所有的数据库实例，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

导出筛选的目标实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，根据引擎类型、实例名称、实例ID、实例内网地址等条件，筛选实例，或勾选需要导出的实例，单击实例列表右上角，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

5.4.5 删除实例

操作场景

您可根据业务需要，在关系型数据库“实例管理”页面手动删除实例来释放资源。

约束限制

- 正在执行操作的实例不能手动删除，只有在实例操作完成后，才可删除实例。
- 实例删除后将不再产生费用，实例生成的自动备份会被同步删除，保留的手动备份会继续收取费用。

须知

- 删除主实例时，会同步删除其对应的备实例和只读实例，请您谨慎操作。
- 实例删除后，资源立即释放，请谨慎操作。如需保留数据，请务必确认完成[数据备份](#)后再删除实例。
- 通过保留的手动备份，可以恢复实例数据，具体请参见[通过备份文件恢复关系型数据库实例数据](#)。

删除主实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择需要删除的主实例，在“操作”列，选择“更多 > 删除实例”。

步骤5 在“删除实例”弹框，单击“是”下发请求。

步骤6 稍后刷新“实例管理”页面，查看删除结果。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择对应主实例，单击实例名称前的，可查看到该主实例下的全部只读实例。

步骤5 选择目标只读实例，在“操作”列，选择“更多 > 删除实例”。

步骤6 在“删除实例”弹框，单击“是”下发请求。

步骤7 稍后刷新“实例管理”页面，查看删除结果。

----结束

5.4.6 回收站

操作场景

RDS支持将删除的主备或者单机实例，加入回收站管理。通过在回收站中重建实例，将数据恢复到新实例上。新实例数据库引擎、数据库版本、存储类型与原实例相同，其他参数可以重新配置。默认可以恢复1~7天内删除的实例。

约束限制

- RDS不回收只读实例，只有主备或者单机实例才会进入回收站。
- 回收站策略机制默认开启，且不可关闭。

设置回收站策略

须知

修改回收站保留天数，仅对修改后新进入回收站的实例生效，对于修改前已经存在的实例，仍保持原来的回收策略，请您谨慎操作。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在左侧导航栏，单击“回收站”。

步骤5 在“回收站”页面，单击“回收站策略”，设置已删除实例保留天数，可设置范围为1~7天。

步骤6 单击“确定”，完成设置。

----结束

重建实例

在回收站保留期限内的主实例可以通过重建实例恢复数据。

步骤1 在左侧导航栏，单击“回收站”。

步骤2 在“回收站”页面，在实例列表中找到需要恢复的目标实例，单击操作列的“重建”。

步骤3 在“重建新实例”页面，选填配置后，提交重建任务，具体可参考[恢复到新实例操作](#)。

----结束

5.5 变更实例

5.5.1 大版本升级

目前不支持通过管理控制台进行大版本升级。您可以使用数据复制服务（Data Replication Service，简称DRS）将云数据库RDS for MySQL 5.6版本迁移到MySQL 5.7版本，平滑完成数据库的迁移工作。**使用该方式进行大版本升级，需要提前准备好待迁移到的高版本数据库实例。**

您可以在“实例管理”页面，单击目标实例名称，在页面右上角，单击“迁移数据库”，进入数据复制服务信息页面。

表 5-4 MySQL 数据库版本信息

源数据库版本	目标数据库版本	迁移类型
RDS for MySQL/自建MySQL/其他云MySQL <ul style="list-style-type: none">● 5.5.x● 5.6.x● 5.7.x● 8.0.x	RDS for MySQL <ul style="list-style-type: none">● 5.6.x● 5.7.x● 8.0.x	MySQL数据库版本升级

说明

DRS仅支持从低版本迁移到高版本。

5.5.2 修改实例名称

操作场景

关系型数据库服务支持修改主实例和只读实例的实例名称，以方便用户识别。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称后的 ，编辑实例名称，单击“确认”，即可修改实例名称。

您也可以单击目标实例名称，进入实例的“基本信息”页面，在“数据库信息”模块实例名称处，单击 ，修改实例名称。

实例名称长度最大64个字符，必须以字母开头，可包含大写字母、小写字母、数字、中划线、下划线，不能包含其他特殊字符。

- 单击 ，提交修改。
- 单击 ，取消修改。

步骤5 在实例的“基本信息”页面，查看修改结果。

----结束

5.5.3 切换主备实例的可用性策略

操作场景

关系型数据库服务支持切换主备实例的可用性策略，以满足不同业务需求。可选择“可靠性优先”或者“可用性优先”两种策略。

- 可靠性优先（默认选择）：数据库应该尽可能保障数据的可靠性，即数据丢失量最少。对于数据一致性要求较高的业务，建议选择该策略。
- 可用性优先：数据库应该尽可能快速恢复服务，即可用时间最长。对于数据库在线时间要求较高的业务，建议选择该策略。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击主实例名称。

步骤5 在“基本信息”页面，在“数据库信息”模块的“切换策略”处，单击“修改”，在弹出框中，选择策略，单击“确定”，完成修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

5.5.4 变更实例的 CPU 和内存规格

操作场景

CPU/内存规格可根据业务需要进行变更，当实例的状态由“规格变更中”变为“正常”，则说明变更成功。

约束限制

- 当实例进行CPU/内存规格变更时，该实例不可被删除。
- 修改CPU/内存后，将会重启数据库实例。请选择业务低峰期，避免业务异常中断。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 规格变更”，进入“规格变更”页面。

您也可以通过单击目标实例名称，进入“基本信息”页面，在“数据库信息”模块的“性能规格”处，单击“规格变更”，进入“规格变更”页面。

步骤5 在“规格变更”页面，选择所需修改的性能规格，单击“下一步”。

步骤6 查看变更结果。

任务提交成功后，单击“返回云数据库RDS列表”，在实例管理页面，可以看到实例状态为“规格变更中”。稍后在对应的“基本信息”页面，查看实例规格，检查修改是否成功。此过程需要5~15分钟。

须知

RDS MySQL实例变更成功后，系统将根据新内存大小，调整如下参数的值：
“innodb_buffer_pool_size”、“innodb_log_buffer_size”、
“innodb_log_files_in_group”、“max_connections”、
“innodb_page_cleaners”、“innodb_buffer_pool_instances”、
“threadpool_size”、“slave_parallel_workers”和“back_log”。

----结束

5.5.5 磁盘扩容

操作场景

随着业务数据的增加，原来申请的数据库磁盘容量可能会不足，需要为关系型数据库实例进行扩容。

1TB以下磁盘空间使用率达到95%，或1TB及以上磁盘空间使用量剩余50GB，实例将显示“磁盘空间满”，此时数据库不可进行写入操作。您需要扩容至磁盘空间使用率小于85%才可使实例处于可用状态，使数据库恢复正常的写入操作。

建议您设置“磁盘使用率”指标的告警规则，及时了解磁盘使用情况，起到预警作用，具体请参见[设置告警规则](#)。

数据库磁盘空间不足的原因和解决方案，请参见[数据超过了RDS实例的最大存储容量怎么办](#)。

磁盘扩容期间，服务不中断。关系型数据库服务只支持扩容磁盘存储空间大小，不支持变更存储类型。

约束限制

- 云数据库MySQL磁盘最大可扩容至4000GB，扩容次数没有限制。如果您想提高存储空间扩容上限到10000GB，请联系技术人员开通。
- 磁盘扩容期间，实例状态为“扩容中”备份业务不受影响。
- 如果是主备实例，针对主实例扩容时，会同时对其备实例进行扩容。
- 扩容过程中，该实例不可重启和删除。
- 磁盘容量变更只允许扩容，不能缩容。

主实例扩容

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 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。

您还可以通过如下途径进行扩容：

- 单击目标实例名称，进入“基本信息”页面。在“存储空间”模块，单击“磁盘扩容”，进入“磁盘扩容”页面。

步骤5 在“磁盘扩容”页面，选择空间大小，单击“下一步”。

每次扩容最小容量为10GB，实例所选容量大小必须为10的整数倍。

步骤6 规格确认。

- 重新选择：单击“上一步”，回到上个页面，修改新增大小。
- 确认无误：单击“提交申请”，提交扩容。
- 确认无误：单击“提交”，提交扩容。

步骤7 查看扩容结果。

在实例管理页面，可看到实例状态为“扩容中”，稍后单击实例名称，在“基本信息”页面，查看磁盘大小，检查扩容是否成功。此过程需要3~5分钟。

针对MySQL，您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。

----结束

只读实例扩容

只读实例磁盘扩容与主实例的磁盘扩容互不影响，因此，您需要单独为只读实例扩容，以满足业务需求。对只读实例进行扩容时，您可选择大于或等于主实例的存储空间。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择相应的实例，单击实例名称前的∨可查看到只读实例，单击“操作”列的“更多 > 磁盘扩容”。

您还可以通过如下途径进行扩容：

- 单击目标实例名称，进入“基本信息”页面。在“存储空间”模块，单击“磁盘扩容”，进入“磁盘扩容”页面。

步骤5 在“服务选型”页面，选择空间大小，单击“下一步”。

每次扩容最小容量为10GB，实例所选容量大小必须为10的整数倍。

步骤6 规格确认。

- 重新选择：单击“上一步”，回到上个页面，修改新增大小。
- 确认无误：单击“提交申请”，提交扩容。
- 确认无误：单击“提交”，提交扩容。

步骤7 查看扩容结果。

在实例管理页面，可看到实例状态为“扩容中”，稍后单击实例名称，在“基本信息”页面，查看磁盘大小，检查扩容是否成功。此过程需要3~5分钟。

针对MySQL，您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。

----结束

5.5.6 设置可维护时间段

操作场景

默认可维护时间段为02:00~06:00，您可以根据业务需求，设置可维护时间段。建议将可维护时间段设置在业务低峰期，避免业务在维护过程中异常中断。

注意事项

- 在可维护时间段内，实例会发生1到2次连接闪断，请确保应用程序具备重连机制。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称，进入“基本信息”页面，在“数据库信息”模块的“可维护时间段”处，单击“修改”。

步骤5 在“修改可维护时间段”弹框中，选择一个可维护时间段，单击“确定”。

说明

修改可维护时间段，不影响原有可维护时间段内定时任务的执行时间。

----结束

5.5.7 单机转主备实例

操作场景

- 关系型数据库支持数据库单机实例转为主备实例。在保留原实例资源的情况下提高了实例的可靠性。单机转主备实例操作对主实例业务没有影响。
- 主备实例可实现自动故障倒换，备机快速接管业务。建议您将主备实例选择到不同的可用区，享受跨可用区，同城容灾的高可用服务。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择单机实例，单击“操作”列的“更多 > 转主备”，进入“转主备”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例左侧的添加按钮 ，进入“转主备”页面。

步骤5 在转主备时，您只需选择“备可用区”，其他信息默认与主实例相同。确认信息无误，单击“立即申请”。

步骤6 单机转主备创建成功后，单击“返回云数据库RDS列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 创建过程中，状态显示为“转主备中”。您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。

- 在实例列表的右上角，单击  刷新列表，可查看到单机转主备完成后，实例状态显示为“正常”，“实例类型”显示为“主备”。

----结束

5.5.8 只读转单机实例

操作场景

关系型数据库服务支持数据库只读实例转为单机实例。它实现了只读实例与主实例断开数据复制关系，成为独立的单机实例。只读转单机实例的操作对主实例业务没有影响。

当主实例故障不可用时，可以通过只读升主，快速获得一个可读可写的单机实例。

说明

- 仅MySQL 5.7和MySQL 8.0版本支持只读实例转单机实例。
- 开启proxy代理的实例不支持转单机实例功能。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 只读升主”，进入“只读升主”页面。

步骤5 只读转单机成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 转换过程中，状态显示为“只读升主中”。
- 在实例列表的右上角，单击  刷新列表，可查看到只读升主完成后，只读实例已脱离原主实例，状态显示为“正常”。“实例类型”显示为“单机”。
- 只读升主后，该实例的计费不变。

---结束

5.5.9 手动切换主备实例

操作场景

当关系型数据库主备类型的实例创建成功后，系统会在同一区域内为该实例创建一个备实例。主备实例的数据会实时同步，用户仅能访问主实例数据，备实例仅作为备份形式存在。您可根据业务需要，进行主备实例的切换，切换后原来的主实例会变成备实例，可实现机架级别的容灾。

约束限制

- 实例运行正常。
- 主备实例复制关系正常。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在“基本信息”页面中“数据库信息”模块的“实例类型”处，单击“主备切换”。

您也可以在“基本信息”页面，单击“实例拓扑图”模块的 。进行主备切换。

须知

主备切换可能会造成几秒或几分钟的服务闪断（闪断时间与复制时延有关），并有可能在主备同步时延过大的情况下，导致少量数据丢失。请在业务低峰期进行主备切换，避免对高峰期业务造成影响。

步骤6 在“主备切换”弹框，单击“是”进行主备实例的切换。

在“复制状态”为“正常”的情况下，如果复制时延大于300s，主备切换任务无法下发。

步骤7 主备切换成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 切换过程中，状态显示为“主备切换中”。
- 在实例列表的右上角，单击刷新列表，可查看到主备切换完成后，实例状态显示为“正常”。

----结束

5.5.10 迁移可用区

操作场景

您可以将主备实例的备机迁移至同一区域内的其它可用区。

说明

- 仅MySQL 5.6和MySQL 5.7版本的主备实例支持可用区迁移功能，其中存储类型是“本地盘SSD”的实例不支持可用区迁移功能。
- 迁移期间将短暂停止DDL语句和event定时任务，请选择业务低峰期操作，避免业务异常中断。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 可用区迁移”，进入“可用区迁移”页面。

步骤5 在“可用区迁移”页面，选择目标可用区，单击“提交”。

步骤6 迁移可用区成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 可用区迁移过程中，状态显示为“备机迁移中”。您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。
- 在实例列表的右上角，单击刷新列表，可查看到可用区迁移完成后，实例状态显示为“正常”。

- 在“基本信息”页面中“数据库信息”模块的“可用区”处，可以查看到备机迁移后所在的可用区。

----结束

5.6 只读实例

5.6.1 只读实例简介

产品简介

目前，云数据库MySQL实例支持只读实例。

在对数据库有少量写请求，但有大量读请求的应用场景下，单个实例可能无法抵抗读取压力，甚至对主业务产生影响。为了实现读取能力的弹性扩展，分担数据库压力，您可以在某个区域中创建一个或多个只读实例，利用只读实例满足大量的数据库读取需求，以此增加应用的吞吐量。

只读实例为单个物理节点的架构（没有备节点），采用MySQL的原生复制功能，将主实例的更改同步到所有只读实例，而且主实例和只读实例之间的数据同步不受网络延时的影响，只读实例跟主实例在同一区域，但可以在不同的可用区。

功能特点

- 只读实例规格可以与主实例不一致，建议只读实例规格大于等于主实例规格，否则易导致只读实例延迟高、负载高等现象。
- 提供系统性能监控。
关系型数据库提供近20个系统性能的监控视图，如磁盘容量、IOPS、连接数、CPU利用率、网络流量等，用户可以轻松查看实例的负载。
- 备份设置：不支持自动备份设置以及手动备份。
- 实例恢复：不支持通过备份文件或任意时间点创建只读实例，不支持通过备份集覆盖只读实例。
- 数据迁移：不支持将数据迁移至只读实例。
- 数据库管理：不支持创建和删除数据库。
- 账号管理：只读实例不提供创建账号权限。

功能限制

1个主实例最多可以创建5个只读实例。

创建和管理只读实例

- [创建单个只读实例](#)
- [管理只读实例](#)

5.6.2 创建只读实例

操作场景

只读实例用于增强主实例的读能力，减轻主实例负载。

关系型数据库单机实例或主备实例创建成功后，您可根据业务需要，创建只读实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击操作列的“创建只读”，进入“服务选型”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例下方的添加按钮 ，创建只读实例。

步骤5 在“服务选型”页面，填写实例相关信息后，单击“立即创建”。

表 5-5 基本信息

参数	描述
当前区域	只读实例默认与主实例在同一区域。
实例名称	实例名称的长度在4~64个字符之间，必须以字母开头，可包含大写字母、小写字母、数字、中划线、下划线，不能包含其他特殊字符。
数据库引擎	默认与主实例的数据库引擎一致，不可更改。
数据库版本	默认与主实例的数据库版本一致，不可更改。
可用区	关系型数据库服务支持在同一个可用区内或者跨可用区部署数据库主实例和只读实例，以提高可靠性。

表 5-6 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。 关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。

参数	描述
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。 只读实例的存储空间大小默认与主实例一致。

表 5-7 网络

参数	描述
虚拟私有云	和主实例相同。
子网	和主实例相同，创建只读实例时RDS会自动为您配置内网地址，您也可输入子网号段内未使用的内网地址，实例创建成功后该内网地址可修改。
安全组	和主实例相同。

步骤6 在“规格确认”页面，进行信息确认。

- 如果需要重新选择，单击“上一步”，回到服务选型页面修改基本信息。
- 信息确认无误，单击“提交”，下发新增只读实例请求。

步骤7 只读实例创建成功后，您可以对其进行查看和管理。

管理只读实例操作，请参见[管理只读实例](#)。

您可以通过“任务中心”查看详细进度和结果。

----结束

后续操作

[管理只读实例](#)

5.6.3 管理只读实例

通过只读实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的∨，单击目标只读实例的名称，进入实例的基本信息页面，即进入只读实例的管理页面。

----结束

通过主实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 单击只读实例所在的主实例名称，进入主实例的“基本信息”页面。

步骤5 在“实例拓扑图”中单击只读实例的名称，进入该只读实例的“基本信息”页面，即进入只读实例的管理页面。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的∨，选择目标只读实例，单击操作列中的“更多 > 删除实例”。

----结束

5.7 备份与恢复

5.7.1 备份概述

关系型数据库支持数据库实例的备份和恢复，以保证数据可靠性。

自动备份

关系型数据库服务会在数据库实例的备份时段中创建数据库实例的自动备份。系统根据您指定的备份保留期保存数据库实例的自动备份。如果需要，您可以将数据恢复到备份保留期中的任意时间点。具体请参见[设置自动备份策略](#)。

手动备份

您还可以创建手动备份对数据库进行备份，手动备份是由用户启动的数据库实例的全量备份，手动备份会一直保存，直到用户手动删除。具体请参见[创建手动备份](#)。

下载备份文件

您可以下载全量备份和增量备份文件，用于本地存储备份或者恢复数据库。具体请参见[下载备份文件](#)和[下载Binlog备份文件](#)。

5.7.2 设置自动备份策略

操作场景

创建关系型数据库实例时，系统默认开启自动备份策略，安全考虑，实例创建成功后不可关闭，您可根据业务需要设置自动备份策略，关系型数据库服务按照您设置的自动备份策略对数据库进行备份。

关系型数据库服务的备份操作是实例级的，而不是数据库级的。当数据库故障或数据损坏时，可以通过备份恢复数据库，从而保证数据可靠性。备份以压缩包的形式存储在对象存储服务上，以保证用户数据的机密性和持久性。由于开启备份会损耗数据库读写性能，建议您选择业务低峰时间段设置自动备份。

关系型数据库默认开启的自动备份策略设置如下：

- 保留天数：默认为7天。
- 备份时间段：默认为24小时中，间隔一小时的随机的一个时间段，例如01:00~02:00，12:00~13:00等。备份时间段以UTC时区保存。如果碰到夏令时/冬令时切换，备份时间段会因时区变化而改变。
- 备份周期：默认为一周内的每一天。

约束限制

- 实例中表的数量会影响备份速度，表数量上限为50万。
- 全量备份时，会连接备份所属的实例，校验该实例的状态。如果校验存在以下两种情况，则校验不通过，会自动进行校验重试。如果重试结束后，仍然无法满足，则备份失败。
 - 备份所属的实例正在执行DDL操作。
 - 从备份所属的实例获取备份锁失败。

查看或修改自动备份策略

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏，单击“备份恢复”，单击“修改备份策略”。您可以查看到已设置的备份策略，如需修改备份策略，请调整以下参数的值。

- 保留天数是指自动备份可保留的时间，增加保留天数可提升数据可靠性，请根据需要设置。
- 减少保留天数的情况下，该备份策略对已有备份文件同时生效，即超出备份保留天数的已有备份文件会被删除。

- 保留天数为全量自动备份和Binlog备份的保留时长，范围为1~732天，备份时间段为间隔1小时，建议根据业务情况，选择业务低峰时段，备份周期默认全选，可修改，且至少选择一周中的一天。

步骤6 单击“确定”，确认修改。

----结束

5.7.3 设置跨区域备份策略

操作场景

关系型数据库支持将备份文件存放到另一个区域存储，某一区域的实例故障后，可以在异地区域使用备份文件在异地恢复到新的RDS实例，用来恢复业务。

实例开启跨区域备份策略后，会自动将该实例的备份文件备份到目标区域。您可以在RDS“备份管理”页面，对跨区域的备份文件进行管理。

开启或修改跨区域备份策略

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏，单击“备份恢复”，单击“设置跨区域备份策略”。

- 开启跨区域全备：将本实例的自动全量备份文件备份到异地。
- 开启跨区域日志备份：将本实例的Binlog备份（增量备份）文件备份到异地。
- 跨区域备份文件的保留天数范围为1~1825天。
- 开启跨区域备份策略仅对新产生的备份生效，即为开启跨区域备份策略后产生的备份生效，之前的备份文件不会存储到异地。
- 开启跨区域日志备份后需要待下一次自动全量备份复制完成后才可进行将数据库实例恢复到指定时间点操作，并且只允许恢复到自动全量备份完成后的时间点。
- 同一个租户下各数据库实例，备份选择的目标区域需要相同。
- 跨区域备份会在实例本区域备份完成后异步复制到异地。

步骤6 单击“确定”，确认修改。

步骤7 您可以在“备份管理”页面的“数据库跨区域备份”页签，管理跨区域备份文件。

- 单击“设置区域备份”，可以修改跨区域备份策略。
- 单击“查看跨区域备份”，可以查看到生成的备份文件，当原区域实例发生故障，可以通过该备份文件将数据恢复到新实例。

----结束

关闭跨区域备份策略

步骤1 在“备份管理”页面，选择“数据库跨区域备份”。

步骤2 在目标实例上，单击“设置区域备份”。在弹出页面，通过设置开关，关闭跨区域备份策略。

步骤3 单击“确定”，确认关闭。

----结束

5.7.4 创建手动备份

操作场景

关系型数据库服务支持对运行正常的主实例创建手动备份，用户可以通过手动备份恢复数据，从而保证数据可靠性。

说明

当数据库实例被删除时，关系型数据库实例的自动备份将被同步删除，手动备份不会被删除。

约束限制

- 实例中表的数量会影响备份速度，表数量上限为50万。
- 全量备份时，会连接备份所属的实例，校验该实例的状态。如果校验存在以下两种情况，则校验不通过，会自动进行校验重试。如果重试结束后，仍然无法满足，则备份失败。
 - 备份所属的实例正在执行DDL操作。
 - 从备份所属的实例获取备份锁失败。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，在操作列选择“更多 > 创建备份”。

步骤5 在创建备份弹出框中，命名该备份，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!"&'=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏，单击“备份恢复”，对其进行查看并管理。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“备份恢复”，单击“创建备份”，命名该备份，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!<"&'=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏中选择“备份恢复”，对其进行查看并管理。

----结束

5.7.5 下载备份文件

操作场景

用户可以下载手动和自动备份文件，用于本地存储备份或者恢复数据库。

云数据库MySQL支持用户下载全量备份。

方式 1：直接浏览器下载

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要下载的可用备份，单击操作列中的“下载”。

您也可进入目标实例的“基本信息”页面，在左侧导航栏选择“备份恢复”，在“全量备份”子页签下，单击操作列中的“下载”。

步骤5 在弹出框中，单击“直接下载”，单击“确定”，通过浏览器直接下载数据库实例的备份文件。

----结束

方式 2：按地址下载

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要下载的可用备份，单击操作列中的“下载”。

您也可进入目标实例的“基本信息”页面，在左侧导航栏选择“备份恢复”，在“全量备份”子页签下，单击操作列中的“下载”。

步骤5 在弹出框中，单击“按地址下载”，单击  复制地址，通过下载地址下载备份文件。

对于MySQL，显示当前备份文件在链接有效期内的下载地址。

- 您可以使用其他下载工具下载备份文件。
- 您还可以使用wget命令下载备份文件：

```
wget -O FILE_NAME --no-check-certificate "DOWNLOAD_URL"
```

命令中的参数解释如下：

FILE_NAME：下载成功后的备份文件名称，由于原始文件名称较长，可能会超出客户端文件系统的限制，建议下载备份文件时使用“-O”进行重命名。

DOWNLOAD_URL：需下载的备份文件所在路径，如果包含特殊字符则需要转义。

----结束

5.7.6 下载 Binlog 备份文件

操作场景

用户可以下载手动和自动备份文件，用于本地存储备份或者恢复数据库。

RDS for MySQL支持用户下载Binlog备份。

下载 Binlog 备份文件

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称，进入“基本信息”页面。

步骤5 在左侧导航栏，单击“备份恢复”，在“Binlog备份”页签下，单击操作列中的“下载”。

您也可以勾选需要下载的Binlog备份，单击左上角“下载”。

步骤6 下载任务执行完成后，您可在本地查看到Binlog备份文件。

----结束

5.7.7 设置 MySQL 本地 Binlog 日志清理

操作背景

RDS for MySQL支持设置本地Binlog日志的清理，所有需要清理的本地日志都会在Binlog备份成功后删除，您可以根据需求灵活设置本地Binlog保留时长，合理利用实例使用空间。

说明

RDS默认开启Binlog，无需额外配置，binlog_format格式为row。

Binlog本地保留时长，可设置范围为0~7*24小时。

清理Binlog日志时，不会清理主备复制所需的Binlog日志，保证主备（包括只读）复制正常，即使设置保留时长为0，RDS也会保留主备复制所需的Binlog日志。

操作步骤

- 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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。
- 步骤5** 在左侧导航栏，单击“备份恢复”，在“Binlog备份”子页签下，单击“Binlog本地设置”。
- 步骤6** 在“Binlog本地设置”弹出框中，设置保留时长，单击“确定”。

图 5-1 Binlog 本地设置



----结束

5.7.8 通过备份文件恢复关系型数据库实例数据

操作场景

关系型数据库服务支持使用已有的自动备份和手动备份，将实例数据恢复到备份被创建时的状态。该操作恢复的为整个实例的数据。

通过备份文件恢复到实例上，会从OBS备份空间中将该备份文件下载到实例上进行全量恢复，恢复时长和实例的数据量有关，平均恢复速率为40MB/s。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在左侧导航栏单击“备份管理”，选择需要恢复的备份，单击操作列的“恢复”。

您也可在“实例管理”页面，单击指定的实例名称，在左侧导航栏单击“备份恢复”，在“全量备份”页签下单击目标备份对应的操作列中的“恢复”。

步骤5 选择需要的恢复方式，单击“确定”。

- **新实例：**将备份恢复到新创建的实例上。
选择恢复到“新实例”，单击“确定”，跳转到“恢复到新实例”的服务选型页面：
 - 数据库引擎和数据库版本，与原实例相同，不可重置。
 - 存储空间大小默认和原实例相同，且必须大于或等于原实例存储空间大小。
 - 其他参数默认，用户如需设置，请参见[步骤一：创建实例](#)。

步骤6 查看恢复结果。

- **恢复到新实例**
为用户重新创建一个和该备份数据相同的实例。可看到实例由“创建中”变为“正常”，说明恢复成功。
恢复成功的新实例是一个独立的实例，与原有实例没有关联。如需使用只读实例，请重新在该实例上进行创建。
新实例创建成功后，系统会自动执行一次全量备份。

---结束

5.7.9 将数据库实例恢复到指定时间点

操作场景

关系型数据库服务支持使用已有的自动备份，恢复实例数据到指定时间点。

实例恢复到指定时间点，会从OBS备份空间中选择一个该时间点最近的全量备份下载到实例上进行全量恢复，再重放增量备份到指定时间点，恢复时长和实例的数据量有关，平均恢复速率为30MB/s。

对当前实例恢复操作

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“备份恢复”页签，单击“恢复到指定时间点”。

步骤6 选择恢复日期和该日期下可恢复的时间区间，并输入要恢复到的时间点，选择恢复方式，单击“确定”。

- 恢复到新实例

跳转到“恢复到新实例”的服务选型页面：

- 数据库引擎和数据库版本，与原实例相同，不可重置。
- 其他参数默认，用户如需设置，请参见[步骤一：创建实例](#)。

步骤7 查看恢复结果。

- 恢复到新实例

为用户重新创建一个和该时间点数据相同的实例。可看到实例由“创建中”变为“正常”，说明恢复成功。

恢复成功的新实例是一个独立的实例，与原有实例没有关联。如需使用只读实例，请重新在该实例上进行创建。

实例创建完成后，系统会自动执行一次全量备份。

----结束

5.7.10 复制备份

操作场景

关系型数据库服务支持复制自动和手动备份，制作一个不同名的拷贝。

限制条件

只支持在同一区域下复制并使用备份，暂不支持跨region操作。

备份保留策略

- 如果需要长期保留自动备份，您可以复制该自动备份生成一个手动备份，该手动备份在您删除之前将会一直保留。
- 如果手动备份所使用的空间超过您的默认存储空间，则可能会增加关系型数据库存储成本。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入“基本信息”页面，在左侧导航栏，单击“备份恢复”，选择目标备份，单击操作列中的“复制”。

步骤5 填写新备份名称和描述，单击“确定”。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，不能包含回车和>!"&'=特殊字符。

步骤6 新备份创建成功后，用户可在“备份管理”页面，查看并管理自己的备份。

----结束

5.8 参数模板管理

5.8.1 创建参数模板

您可以使用数据库参数模板中的参数来管理数据库引擎配置。数据库参数模板就像是引擎配置值的容器，这些值可应用于一个或多个数据库实例。

默认参数模板包含数据库引擎默认值和系统默认值，具体根据引擎、计算等级及实例的分配存储空间而定。您无法修改默认数据库参数模板的参数设置，您必须创建自己的数据库参数模板才能更改参数设置的默认值。

须知

并非所有数据库引擎参数都可在客户创建的数据库参数模板中进行更改。

如果您想使用您自己的数据库参数模板，只需创建一个新的数据库参数模板，创建实例的时候选择该参数模板，如果是在创建实例后有这个需求，可以重新应用该参数模板，请参见[应用参数模板](#)。

若您已成功创建数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案，请参见[复制参数模板](#)。

以下是您在使用数据库参数模板中的参数时应了解的几个要点：

- 当您更改动态参数并保存数据库参数模板时，将立即应用更改。当您更改静态参数并保存数据库参数模板时，参数更改将在您手动重启数据库实例后生效。
- 在数据库参数模板内设置参数不恰当可能会产生意外的不利影响，包括性能降低和系统不稳定。修改数据库参数时应始终保持谨慎，且修改数据库参数模板前要备份数据。将参数模板更改应用于生产数据库实例前，您应当在测试数据库实例上试用这些参数模板设置更改。

说明

关系型数据库和文档数据库服务不共享参数模板配额。

每个用户最多可以创建100个关系型数据库参数模板，各关系型数据库引擎共享该配额。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，单击“创建参数模板”。

步骤5 选择数据库引擎版本，命名参数模板并添加对该参数模板的描述，单击“确定”，创建参数模板。

- 选择该数据库引擎参数模板所需应用的参数模板类型。
- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<"&'=特殊字符。

---结束

5.8.2 编辑参数模板

为确保关系型数据库服务发挥出最优性能，用户可根据业务需求对用户创建的参数模板中的参数进行调整。

您可以修改用户创建的数据库参数模板中的参数值，但不能更改默认数据库参数模板中的参数值。

全局参数必须通过控制台修改，会话级参数可通过SQL修改。如果您更改一个参数值，则所做更改的应用时间将由该参数的类型决定。

关系型数据库服务的管理控制台显示与数据库参数模板关联的数据库实例状态。例如，如果数据库实例未使用对其关联的数据库参数模板所做的最新更改，则关系型数据库服务的管理控制台将显示状态为“等待重启”的数据库实例。您将需要手动重启数据库实例，以使最新的参数更改对该数据库实例生效。

说明

系统提供的默认参数模板不允许修改，只可单击参数模板名进行查看。当用户参数设置不合理导致数据库无法启动时，可参考默认参数模板重新配置。

批量修改参数

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 > 关系型数据库”。进入关系型数据库信息页面。

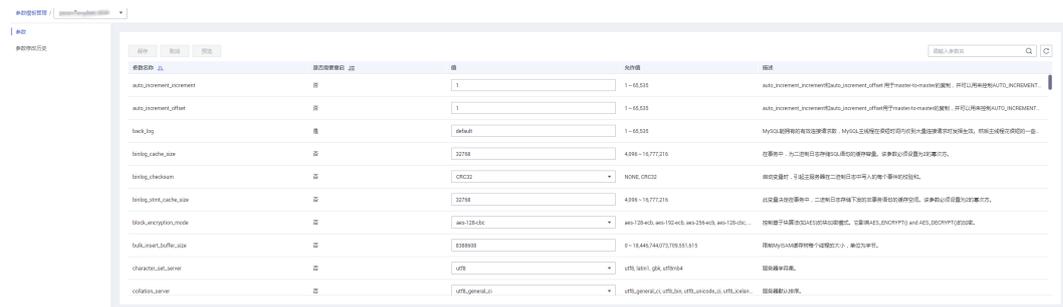
步骤4 在“参数模板管理”页面的“自定义”页签，选择目标参数模板，单击参数模板名称。

步骤5 默认在“参数”页签下，根据需要修改相关参数值。

参数相关说明，请参见[MySQL参数调优建议](#)。

可进行的操作如下：

图 5-2 编辑参数模板



- 单击“保存”，在弹出框中单击“是”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

步骤6 参数修改完成后，您可以单击“参数修改历史”查看参数的修改详情。

步骤7 参数模板修改后，不会立即应用到当前使用的实例，您需要应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤8 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

----结束

修改当前实例的参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”页签修改相应参数。

须知

根据参数列表中“是否需要重启”提示，进行相应操作：

- 是：在实例列表中，查看“运行状态”，如果显示“参数变更，等待重启”，则需重启实例使之生效。
 - 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
 - 修改只读实例的某些参数，需要重启该只读实例使之生效。
- 否：无需重启，立即生效。

- 单击“保存”，在弹出框中单击“是”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

参数修改完成后，您可在“参数修改”页面，选择“参数修改历史”页签查看参数的修改详情。

----结束

5.8.3 导出参数模板

操作场景

- 您可以将您创建的数据库实例参数列表，导出并生成一个新的参数模板，供您后期使用。请参考[应用参数模板](#)将导出的参数模板应用到新的实例。
- 您可以将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV中，方便查看并分析。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”页签单击“导出”。

- 导出到参数模板。将该实例对应参数列表导出并生成一个参数模板，供您后期使用。
在弹出框中，填写新参数模板名称和描述，单击“确定”。

说明

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<'&=特殊字符。

创建完成后，会生成一个新的参数模板，您可在“参数模板管理”页面的对其进行管理。

- 导出到文件。将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV表中，方便用户查看并分析。
在弹出框中，填写文件名称，单击“确定”。

📖 说明

文件名称长度在4~64个字符之间，必须以字母开头，可以包含字母、数字、中划线或下划线，不能包含其他特殊字符。

----结束

5.8.4 比较参数模板

操作场景

您可以比较同数据库类型的实例参数模板，以了解当前实例参数的差异项。

您也可以比较同数据库类型的默认参数模板，以了解当前参数模板的配置情况。

比较当前实例参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”子页签中单击“比较参数”。

步骤6 在弹出框中选择当前实例同数据库类型的参数模板，单击“确定”，比较两个参数的差异项。

- 有差异项，则会显示差异参数的如下信息：参数名称、当前实例参数模板的参数值和被比较参数模板的参数值。
- 无差异项，则不显示。

----结束

比较目标参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“比较”。

步骤5 选择同一数据库引擎的不同参数模板，单击“确定”，比较两个参数模板之间的配置参数差异项。

- 有差异项，则会显示差异参数模板的如下信息：参数名称、两个参数模板的参数值。

- 无差异项，则不显示。

----结束

5.8.5 查看参数修改历史

操作场景

您可以查看当前实例所使用参数模板以及自定义参数模板的修改历史，以满足业务需要。

说明

用户创建或导出的新参数模板，在未进行参数修改前，无修改历史。

查看当前实例的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏，单击“参数修改”，单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态、修改时间、是否应用以及应用时间。

如果修改后参数模板未应用，请根据业务需要，参考[应用参数模板](#)，将其应用到对应实例。

----结束

查看目标参数模板的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，单击目标参数模板名称。

步骤5 单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态和修改时间。

----结束

5.8.6 复制参数模板

操作场景

您可以复制您创建的自定义数据库参数模板。当您已创建一个数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案。

复制数据库参数模板之后，新参数模板可能不会立即显示，建议您等待5分钟再使用。

您无法复制默认参数模板。不过，您可以创建基于默认参数模板的新参数模板。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要复制的参数模板，单击“复制”。

您还可以在“实例管理”页面，单击实例名称，在左侧导航栏，单击“参数修改”，单击“导出”，将该实例对应参数列表导出并生成一个参数模板，供您后期使用。

步骤5 在弹出框中，填写新参数模板名称和描述，单击“确定”。

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<''&'=特殊字符。

创建完成后，会生成一个新的参数模板，您可在参数模板列表中对其进行管理。

----结束

5.8.7 重置参数模板

操作场景

您可根据自己的业务需求，重置自己创建的参数模板对应的所有参数，使其恢复到默认值。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要设置的参数模板，单击“更多 > 重置”。

步骤5 单击“是”，重置所有参数为其默认值。

步骤6 参数模板修改后，不会立即应用到当前使用的实例，您需要进行应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤7 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

----结束

5.8.8 应用参数模板

操作场景

参数模板编辑修改后，您可以根据业务需要应用到实例中。

- 参数“innodb_buffer_pool_size”跟内存强相关，不同规格的实例有不同的区间范围，如果应用参数模板时，该参数超过了实例本身的区间大小，则会取实例区间范围的最大值。
- 参数模板只能应用于相同版本的实例中。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，根据参数模板类型不同进行如下操作。

- 若需要将默认参数模板应用到实例，在“系统默认”页签的目标参数模板单击“应用”。
- 若需要将用户自己创建的参数模板应用到实例，在“自定义”页签的目标参数模板单击“更多 > 应用”。

一个参数模板可被应用到一个或多个实例。

步骤5 在弹出框中，选择或输入所需应用的实例，单击“确定”。

参数模板应用成功后，您可[查看参数模板应用记录](#)。

----结束

5.8.9 查看参数模板应用记录

操作场景

参数模板编辑修改后，您可根据业务需要将其应用到对应实例中，RDS支持查看参数模板所应用到实例的记录。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 单击“参数模板管理”。

步骤5 在“系统默认”页签下，选择目标参数模板，单击“应用记录”；或在“自定义”页签下，选择目标参数模板，单击“更多 > 应用记录”，查看应用记录。

您可查看参数模板所应用到的实例名称/ID、应用状态、应用时间、失败原因。

----结束

5.8.10 修改参数模板描述

操作场景

参数模板创建成功后，用户可根据需要对自己创建的参数模板描述进行修改。

说明

默认参数模板的描述不可修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“描述”列 。

步骤5 输入新的描述信息，单击 ，提交修改，单击，取消修改。

- 参数模板的描述长度不能超过256个字符，且不能包含>!<"&'=特殊字符。
- 修改成功后，可在参数模板列表的“描述”列查看改后的描述信息。

----结束

5.8.11 删除参数模板

操作场景

您可删除废弃的参数模板。

须知

- 参数模板删除后，不可恢复，请谨慎操作。
- 默认参数模板不可被删除。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要删除的参数模板，单击“更多 > 删除”。

步骤5 单击“是”，删除参数模板。

----结束

5.9 连接管理

5.9.1 配置和修改内网地址

操作场景

用户从线下或者其他云迁移到关系型数据库后要面对更改IP的问题，为减少客户业务更改，降低迁移难度。提供规划与更改内网IP方式，降低客户迁移成本。

约束限制

修改内网IP后域名需要几分钟重新解析地址导致数据库连接中断，请在业务停止期间操作。

配置内网 IP

在创建实例时，可在“服务选型”页面的网络部分，根据选择的子网自动配置内网地址。

修改内网 IP

对于创建完成的关系型数据库实例，支持更改内网地址。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 在“基本信息”页“连接信息”模块“内网地址”处，单击“修改”。

步骤6 在“修改内网地址”弹出框中进行修改。单击“确定”，保存修改内容。

已使用IP地址列表中使用情况为“待使用”的IP地址表示已被占用，不允许被当前实例使用。

----结束

5.9.2 绑定和解绑弹性公网 IP

操作场景

关系型数据库实例创建成功后，默认未开启公网访问功能（即未绑定弹性公网IP）。关系型数据库服务支持用户绑定弹性公网IP，在公共网络来访问数据库实例，绑定后也可根据需要解绑。

须知

为保证数据库可正常访问，请确保数据库使用的安全组开通了相关端口的访问权限，假设数据库的访问端口是8635，那么需确保安全组开通了8635端口的访问。

前提条件

- 用户需要在VPC申请一个弹性公网IP。
- 只有主实例和只读实例才能绑定弹性公网IP。
- 对于已绑定弹性公网IP的实例，需解绑后，才可重新绑定其他弹性公网IP。

绑定弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 在左侧导航栏，选择“弹性公网IP”，单击“绑定弹性公网IP”。

步骤6 在弹出框的弹性公网IP地址列表中，显示“未绑定”状态的弹性公网IP，选择所需绑定的弹性公网IP，单击“确定”，提交绑定任务。如果没有可用的弹性公网IP，单击“查看弹性公网IP”，获取弹性公网IP。

步骤7 在“弹性公网IP”页面，查看绑定成功的弹性公网IP。

您也可以在“任务中心”页面，查看绑定弹性公网IP任务的执行进度及结果。

如需关闭，请参见[解绑弹性公网IP](#)。

----结束

解绑弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 对于已绑定EIP的实例，在“实例管理”页面，选择指定实例，单击实例名称，进入实例基本信息页面。

步骤5 选择“弹性公网IP”页面，单击IP地址后面的“解绑”，在弹出框中单击“是”，解绑EIP。

步骤6 在“弹性公网IP”页面，查看结果。

您也可以在“任务中心”页面，查看解绑弹性公网IP任务的执行进度及结果。

如需重新绑定，请参见[绑定弹性公网IP](#)。

----结束

5.9.3 修改数据库端口

操作场景

关系型数据库服务支持修改主实例和只读实例的数据库端口，对于主备实例，修改主实例的数据库端口，该实例下备实例的数据库端口会被同步修改。

约束条件

端口修改中，以下操作不可进行：

- 绑定弹性公网IP。
- 删除实例。
- 创建备份。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击主实例名称，或单击，单击只读实例名称。

步骤5 在“基本信息”页面，在“连接信息”模块“数据库端口”处，单击，修改数据库端口。

说明

MySQL数据库端口设置范围为2100~9500。

- 单击 ，提交修改。
 - 在弹出框中，单击“是”，提交修改。
 - i. 修改主实例数据库端口，对应的主备实例均会被修改且重启。
 - ii. 修改只读实例数据库端口，仅修改并重启该只读实例。
 - iii. 此过程需要1~5分钟左右。
 - 在弹出框中，单击“取消”，取消本次修改。
- 单击 ，取消修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

5.9.4 下载证书

RDS for MySQL支持下载证书。

下载证书

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称，进入“基本信息”页面。

步骤5 在“数据库信息”模块的“SSL”处，单击 ，下载根证书或捆绑包。

说明

- 关系型数据库服务在2017年4月提供了20年有效期的新根证书，该证书在实例重启后生效。请在原有根证书到期前及时更换正规机构颁发的证书，提高系统安全性。

请参见[如何确认SSL根证书的有效期](#)。

- 关系型数据库服务还提供根证书捆绑包下载，其中包含2017年4月之后的新根证书和原有根证书。

----结束

5.9.5 设置安全组规则

操作场景

安全组是一个逻辑上的分组，为同一个虚拟私有云内具有相同安全保护需求，并相互信任的弹性云服务器和关系型数据库实例提供访问策略。

为了保障数据库的安全性和稳定性，在使用关系型数据库实例之前，您需要设置安全组，开通需访问数据库的IP地址和端口。

- 通过弹性公网IP连接RDS实例时，需要为RDS所在安全组配置相应的**入方向规则**。
- 通过内网连接RDS实例时，设置安全组分为以下两种情况：
 - ECS与RDS实例在相同安全组时，默认ECS与RDS实例互通，无需设置安全组规则。

- ECS与RDS实例在不同安全组时，需要为RDS和ECS分别设置安全组规则。
 - 设置RDS安全组规则：为RDS所在安全组配置相应的入方向规则。
 - 设置ECS安全组规则：安全组默认规则为出方向上数据报文全部放行，此时，无需对ECS配置安全组规则。当在ECS所在安全组为非默认安全组且出方向规则非全放通时，需要为ECS所在安全组配置相应的出方向规则。

注意事项

因为安全组的默认规则是在出方向上的数据报文全部放行，同一个安全组内的弹性云服务器和关系型数据库实例可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当关系型数据库实例加入该安全组后，即受到这些访问规则的保护。

- 默认情况下，一个租户可以创建100个安全组。
- 默认情况下，一个安全组最多只允许拥有50条安全组规则。
- 目前一个RDS实例仅允许绑定一个安全组。
- 为一个安全组设置过多的安全组规则会增加首包延时，因此，建议一个安全组内的安全组规则不超过50条。
- 当需要从安全组外访问安全组内的关系型数据库实例时，需要为安全组添加相应的入方向规则。

说明

源地址默认的IP地址0.0.0.0/0是指允许所有IP地址访问安全组内的关系型数据库实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 设置安全组规则。

- 在“基本信息”页面中，在“连接信息”模块的“安全组”处，单击安全组名称，进入安全组页面。

图 5-3 连接信息



步骤6 在入方向规则页签，单击“添加规则”，添加入方向规则。

单击“+”可以依次增加多条入方向规则。

图 5-4 添加入方向规则



表 5-8 入方向参数说明

参数	说明	取值样例
协议端口	网络协议。目前支持“All”、“TCP”、“UDP”、“ICMP”和“GRE”等协议。	Custom TCP
	端口：允许远端地址访问弹性云服务器指定端口。	通过内网连接实例时 ，输入已购买的弹性云服务器的目标实例的端口。
源地址	源地址：可以是IP地址、安全组。 例如： <ul style="list-style-type: none"> • xxx.xxx.xxx.xxx/32（IPv4地址） • xxx.xxx.xxx.0/24（子网） • 0.0.0.0/0（任意地址） 	0.0.0.0/0
描述	安全组规则的描述信息，非必填项。 描述信息内容不能超过255个字符，且不能包含“<”和“>”。	-

---结束

5.10 数据库管理

5.10.1 创建数据库

操作场景

关系型数据库实例创建成功后，您可根据业务需要，创建更多数据库。

约束限制

- 恢复中的实例，不可进行该操作。
- 数据库管理功能目前仅支持MySQL引擎。

通过 RDS 创建数据库

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页。

步骤5 在左侧导航栏，单击“数据库管理”，单击“创建数据库”，在弹出框中输入数据库名称、备注、选择字符集并授权数据库账号，单击“确定”。

- 数据库名称长度可在1~64个字符之间，由字母、数字、中划线、下划线或\$组成，中划线和\$累计总长度小于等于10个字符，且不能包含其他特殊字符（MySQL 8.0不可包含\$）。
- 字符集默认utf8，您可根据业务进行选择，单击“展开”可查看更多。
- 选择指定的一个或多个未授权账号，单击  进行授权（必选），或选择一个或多个指定的已授权账号，单击  取消授权。

如果没有未授权账号，您可进行创建，具体操作请参见[创建数据库账号](#)。

步骤6 数据库创建成功后，您可在当前实例的数据库列表中，对其进行管理。

须知

“AUTO_PK_ROW_ID”列名是RDS for MySQL数据库保留列名，用户无法创建。

----结束

5.10.2 数据库授权

操作场景

您可将自己创建的数据库账号，授权给指定的数据库，对于已授权的数据库账号，您也可取消授权。

限制条件

恢复中的实例，不可进行该操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页。

步骤5 在左侧导航栏，单击“数据库管理”，选择目标数据库，单击操作列的“授权”。

步骤6 在弹出框中选择一个或多个未授权的数据库账号，单击  进行授权，或选择一个或多个指定的已授权数据库账号，单击  取消授权。

如果没有未授权数据库，您可参见[创建数据库账号](#)进行创建。

步骤7 在弹出框中，单击“确定”，提交修改。

----结束

5.10.3 删除数据库

操作场景

您可删除自己创建的数据库。

须知

数据库一旦删除，数据会丢失，请谨慎操作。

限制条件

恢复中的实例，不可进行该操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页。

步骤5 在左侧导航栏，单击“数据库管理”，选择目标数据库，单击操作列的“删除”，在弹出框中单击“是”，删除数据库。

----结束

5.10.4 开启或关闭事件定时器

操作场景

事件定时器Event Scheduler是事件（event）调度任务的总开关。由于原生事件定时器不能保证主、备库的event状态一致，一旦进行主备切换会导致event调度失败。云数据库MySQL提供了event状态同步功能，确保相关的event正常调度。该功能依赖在console上开启、关闭事件定时器来实现。

- 创建实例时，事件定时器默认不开启。
- 主、备实例切换后，事件状态同步是否开启保持不变，依然保持主库“event_scheduler”为“on”，备库为“off”。
- 恢复到新实例时，事件状态与原实例保持一致。
- 单机实例转为主备实例，事件状态与主实例保持一致。

约束条件

- 仅支持MySQL内核5.6.43.2、5.7.25.2及其以上版本。
- 不支持只读实例开启此功能。

开启事件定时器功能

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击主实例名称。

步骤5 在“基本信息”页面，在“数据库信息”模块的“事件定时器”处，单击。

须知

开启事件定时器后，请登录到该实例，确保要开启的event状态设置为enable。

----结束

关闭事件定时器功能

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击主实例名称。

步骤5 在“基本信息”页面，在“数据库信息”模块的“事件定时器”处，单击 。

----结束

5.11 账号管理（非管理员权限）

5.11.1 创建数据库账号

操作场景

创建关系型数据库实例时，系统默认同步创建root用户，您可根据业务需要，添加其他用户。

限制条件

恢复中的实例，不可进行该操作。

通过 RDS 创建账号

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页。

步骤5 在左侧导航栏，单击“账号管理”，单击“创建账号”。在“创建账号”弹出框中，输入账号名称、授权数据库，并输入密码和确认密码，单击“确定”。

- 数据库账号名称在1到32个字符之间，由字母、数字、中划线或下划线组成，不能包含其他特殊字符。
 - 若数据库版本为MySQL5.6，账号长度为1~16个字符。
 - 若数据库版本为MySQL5.7和8.0，账号长度为1~32个字符。

- 选择一个或多个指定的未授权数据库，单击  进行授权，或选择一个或多个指定的已授权数据库，单击  取消授权。

如果没有未授权数据库，您可进行创建，请参见[创建数据库](#)，后期也可修改权限，请参见[修改数据库账号权限](#)。

- 密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*_=-+?,特殊字符。

步骤6 数据库账号添加成功后，您可在当前实例的数据库账号列表中，对其进行管理。

----结束

5.11.2 重置数据库账号密码

操作场景

您可重置自己创建的数据库账号密码，安全考虑，请定期（如三个月或六个月）修改密码，防止出现密码被暴力破解等安全风险。

限制条件

恢复中的实例，不可进行该操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页。

步骤5 在左侧导航栏，单击“账号管理”，选择目标账号，单击操作列的“重置密码”。

步骤6 在弹出框中输入新密码和确认密码，单击“确定”，提交修改。

- 密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*_=-+?,特殊字符。
- 新密码和确认密码需相同。
- 重置密码后数据库不会重启，权限不会发生变化。

----结束

5.11.3 修改数据库账号权限

操作场景

您可将自己创建的数据库账号，授权给指定的数据库，对于已授权的数据库，您也可取消授权。

限制条件

恢复中的实例，不可进行该操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在左侧导航栏，单击“账号管理”，选择目标账号，单击操作列的“修改权限”。

步骤6 在弹出框中，选择一个或多个指定的未授权数据库，单击  进行授权。您还可以选择一个或多个指定的已授权数据库，单击  取消授权。

- 如果没有未授权数据库，您可参见[创建数据库](#)进行创建。

步骤7 单击“确定”，提交修改。

----结束

5.11.4 删除数据库账号

操作场景

您可删除自己创建的数据库账号。

须知

数据库账号删除后不可恢复，请谨慎操作。

限制条件

恢复中的实例，不可进行该操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在左侧导航栏，单击“账号管理”，选择目标账号，单击操作列的“删除”。

步骤6 在弹出框中单击“是”，提交删除任务。

----结束

5.12 数据库安全设置

账户密码等级设置

关系型数据库对在客户端新创建的数据库用户，设置了密码安全策略：

- 口令长度至少8个字符。
- 口令至少包含大写字母、小写字母、数字和特殊字符各一个。

创建实例时，为用户提供了密码复杂度校验，由于root用户可以修改密码复杂度，安全起见，建议修改后的密码复杂度不低于关系型数据库的初始化设置。

账户说明

您在创建MySQL数据库实例时，系统会自动为实例创建如下系统账户（用户不可使用），用于给数据库实例提供完善的后台运维管理服务。

须知

删除、重命名、修改这些账户的密码和权限信息，会导致实例运行异常，请谨慎操作。

- rdsAdmin：管理账户，拥有最高的superuser权限，用于查询和修改实例信息、故障排查、迁移、恢复等操作。
- rdsRepl：复制账户，用于备实例或只读实例在主实例上同步数据。
- rdsBackup：备份账户，用于后台的备份。
- rdsMetric：指标监控账户，用于watchdog采集数据库状态数据。
- rdsProxy：数据库代理账户，该账户在开通读写分离时才会自动创建，用于通过读写分离地址连接数据库时鉴权使用。

5.13 数据安全性

5.13.1 重置管理员密码

操作场景

关系型数据库服务仅支持通过主实例重置管理员密码。

在使用RDS过程中，如果忘记数据库账号密码，可以重新设置密码。

注意事项

- 当您修改数据库主实例的密码时，如果该实例中存在备实例或只读实例，则会被同步修改。
- 重置密码生效时间取决于该主实例当前执行的业务数据量。
- 请定期修改用户密码，以提高系统安全性，防止出现密码被暴力破解等安全风险。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，选择“更多 > 重置密码”。

步骤5 在“重置密码”弹框，输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@\$%^*_+=?;特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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”页签，在“数据库信息”模块的“管理员账户名”处，单击“重置密码”。

步骤6 输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@\$%^*_+=?;特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

5.13.2 修改实例安全组

操作场景

关系型数据库服务支持修改数据库引擎的主实例和只读实例的安全组，对于主备实例，修改主实例的安全组，备实例的安全组会被同步修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择对应的主实例或只读实例，单击实例名称。

步骤5 在“基本信息”页签“连接信息”模块的“安全组”处，单击 ，选择对应的安全组。

- 单击 ，提交修改。
- 单击 ，取消修改。

步骤6 稍后单击“基本信息”页面右上角的 ，查看修改结果。此过程需1~3分钟。

----结束

5.14 监控指标与告警

5.14.1 支持的监控指标

声明：RDS实例中的Agent只用于监控实例运行的指标、状态，不会收集除监控指标外的其它数据。

功能说明

本节定义了关系型数据库上报云监控的监控指标的命名空间，监控指标列表和维度定义。用户可以通过云监控提供的API接口来检索关系型数据库产生的监控指标和告警信息。

命名空间

SYS.RDS

实例监控指标

MySQL数据库性能监控指标，如表5-9所示。

表 5-9 数据库性能监控列表

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds001_cpu_util	CPU使用率	该指标用于统计测量对象的CPU使用率，以比率为单位。	0-100 %	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds002_mem_util	内存使用率	该指标用于统计测量对象的内存使用率，以比率为单位。	0-100 %	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds003_iops	IOPS	该指标用于统计当前实例，单位时间内系统处理的I/O请求数量（平均值）。	≥ 0 counts/s	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds004_bytes_in	网络输入吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输入的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds005_bytes_out	网络输出吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输出的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds006_conn_count	数据库总连接数	该指标用于统计试图连接到MySQL服务器的总连接数，以个为单位。	≥ 0 counts	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds007_conn_active_count	当前活跃连接数	该指标用于统计当前打开的连接的数量，以个为单位。	≥ 0 counts	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds008_qps	QPS	该指标用于统计SQL语句查询次数，含存储过程，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds009_tps	TPS	该指标用于统计事务执行次数，含提交的和回退的，以次/秒为单位。	≥ 0 transactions/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds010_innodb_buf_usage	缓冲池利用率	该指标用于统计空闲的页与InnoDB缓存中缓冲池页面总数的比例，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds011_innodb_buf_hit	缓冲池命中率	该指标用于统计读命中与读请求数比例，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds012_innodb_buf_dirty	缓冲池脏块率	该指标用于统计InnoDB缓存中脏数据与InnoDB缓存中使用的页比例，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds013_innodb_reads	InnoDB读取吞吐量	该指标用于统计InnoDB平均每秒读字节数，以字节/秒为单位。	≥ 0 bytes/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds014_innodb_writes	InnoDB写入吞吐量	该指标用于统计InnoDB平均每秒写字节数，以字节/秒为单位。	≥ 0 bytes/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds015_innodb_read_count	InnoDB文件读取频率	该指标用于统计InnoDB平均每秒从文件中读的次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds016_innodb_write_count	InnoDB文件写入频率	该指标用于统计InnoDB平均每秒向文件中写的次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds017_innodb_log_write_req_count	InnoDB日志写请求频率	该指标用于统计平均每秒的日志写请求数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds018_innodb_log_write_count	InnoDB日志物理写频率	该指标用于统计平均每秒向日志文件的物理写次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds019_innodb_log_fsync_count	InnoDB日志fsync()写频率	该指标用于统计平均每秒向日志文件完成的fsync()写数量，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds021_mysqlam_buf_usage	Key Buffer利用率	该指标用于统计MyISAM Key buffer的利用率，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds022_mysqlam_buf_write_hit	Key Buffer写命中率	该指标用于统计MyISAM Key buffer写命中率，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds023_mysqlam_buf_read_hit	Key Buffer读命中率	该指标用于统计MyISAM Key buffer读命中率，以比率为单位。	0-1	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds024_mysqlam_disk_write_count	MyISAM硬盘写入频率	该指标用于统计向磁盘写入索引的次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds025_mysqlam_disk_read_count	MyISAM硬盘读取频率	该指标用于统计从磁盘读取索引的次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds026_mysqlam_buf_write_count	MyISAM缓冲池写入频率	该指标用于统计向缓冲池写入索引的请求次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds027_myisam_buf_read_count	MyISAM缓冲池读取频率	该指标用于统计从缓冲池读取索引的请求次数，以次/秒为单位。	≥ 0 counts/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds028_cmdml_delete_count	Delete语句执行频率	该指标用于统计平均每秒Delete语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds029_cmdml_insert_count	Insert语句执行频率	该指标用于统计平均每秒Insert语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds030_cmdml_insert_select_count	Insert_Select语句执行频率	该指标用于统计平均每秒Insert_Select语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds031_cmdml_replace_count	Replace语句执行频率	该指标用于统计平均每秒Replace语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds032_cmdml_replace_select_count	Replace_Selection语句执行频率	该指标用于统计平均每秒Replace_Selection语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds033_cmdml_select_count	Select语句执行频率	该指标用于统计平均每秒Select语句执行次数。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds034_cmdml_update_count	Update语句执行频率	该指标用于统计平均每秒Update语句执行次数，以次/秒为单位。	≥ 0 queries/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds035_in nodb_del_ row_count	行删除速率	该指标用于统计平均每秒从InnoDB表删除的行数，以行/秒为单位。	≥ 0 rows/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds036_in nodb_ins_ row_count	行插入速率	该指标用于统计平均每秒向InnoDB表插入的行数，以行/秒为单位。	≥ 0 rows/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds037_in nodb_read_ _row_cou nt	行读取速率	该指标用于统计平均每秒从InnoDB表读取的行数，以行/秒为单位。	≥ 0 rows/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds038_in nodb_upd_ _row_cou nt	行更新速率	该指标用于统计平均每秒向InnoDB表更新的行数，以行/秒为单位。	≥ 0 rows/s	测量对象：数据库 监控实例类型：MySQL实例	1分钟
rds039_dis k_util	磁盘利用率	该指标用于统计测量对象的磁盘利用率，以比率为单位。	0-100 %	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds047_dis k_total_siz e	磁盘总大小	该指标用于统计测量对象的磁盘总大小。	40GB~ 4000G B	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds048_dis k_used_siz e	磁盘使用量	该指标用于统计测量对象的磁盘使用大小。	0GB~4 000GB	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds049_dis k_read_thr oughput	硬盘读吞吐量	该指标用于统计每秒从硬盘读取的字节数。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟
rds050_dis k_write_th roughput	硬盘写吞吐量	该指标用于统计每秒写入硬盘的字节数。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：MySQL实例	1分钟

维度

Key	Value
rds_instance_id	MySQL实例ID

5.14.2 设置告警规则

操作场景

- 云监控界面支持[设置告警规则](#)，用户可自定义监控目标与通知策略，及时了解关系型数据库服务运行状况，从而起到预警作用。

设置关系型数据库服务的告警规则包括设置告警规则名称、资源类型、维度、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。

设置告警规则

步骤1 登录管理控制台。

步骤2 选择“管理与监管 > 云监控服务 CES”。

步骤3 在左侧导航树栏，选择“告警 > 告警规则”。

步骤4 在“告警规则”界面，单击“创建告警规则”进行添加。

----结束

5.14.3 查看监控指标

操作场景

云服务平台提供的云监控，可以对关系型数据库的运行状态进行日常监控。您可以通过管理控制台，直观地查看关系型数据库的各项监控指标。

由于监控数据的获取与传输会花费一定时间，因此，云监控显示的是当前时间5~10分钟前的关系型数据库状态。如果您的关系型数据库刚创建完成，请等待5~10分钟后查看监控数据。

前提条件

- 关系型数据库正常运行。
故障、删除状态的关系型数据库，无法在云监控中查看其监控指标。当关系型数据库再次启动或恢复后，即可正常查看。

说明

故障24小时及以上的关系型数据库，云监控将默认该关系型数据库不存在，并在监控列表中删除，不再对其进行监控，但告警规则需要用户手动清理。

- 关系型数据库已正常运行一段时间（约10分钟）。
对于新创建的关系型数据库，需要等待一段时间，才能查看上报的监控数据和监控视图。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击操作列中的“查看监控指标”，跳转到云监控服务页面。

您也可以在“实例管理”页面，单击目标实例名称，在页面右上角，单击“查看监控指标”，跳转到云监控服务页面。

步骤5 在云监控页面，可以查看实例监控信息。

云监控支持的性能指标监控时间窗包括：近1小时、近3小时、近12小时、近24小时、近7天。

---结束

5.15 CTS 审计

5.15.1 支持审计的关键操作列表

通过，您可以记录与关系型数据库实例相关的操作事件，便于日后的查询、审计和回溯。

表 5-10 支持的 RDS 操作列表

操作名称	资源类型	事件名称
创建实例、创建只读、恢复到新实例	instance	createInstance
扩容、规格变更	instance	instanceAction
实例重启	instance	instanceRestart
恢复到原有实例	instance	instanceRestore
实例重命名	instance	instanceRename
重置密码	instance	resetPassword
设置数据库版本配置参数	instance	setDBParameters
重置实例的数据库版本配置参数	instance	resetDBParameters
设置备份策略-打开，关闭，修改	instance	setBackupPolicy
修改数据库端口号	instance	changeInstancePort
绑定解绑EIP	instance	setOrResetPublicIP
修改安全组	instance	modifySecurityGroup

操作名称	资源类型	事件名称
创建标签	instance	createTag
删除标签	instance	deleteTag
修改标签	instance	modifyTag
删除实例	instance	deleteInstance
主备倒换	instance	instanceFailOver
修改主备同步模式	instance	instanceFailOverMode
修改主备倒换策略	instance	instanceFailOverStrategy
单机转主备实例	instance	modifySingleToHaInstance
创建快照	backup	createManualSnapshot
复制快照	backup	copySnapshot
下载备份（通过OBS下载）	backup	downloadSnapshot
下载备份（通过浏览器下载）	backup	backupsDownload
删除快照	backup	deleteManualSnapshot
合并下载binlog	backup	packBackupsDownload
创建参数模板	parameterGroup	createParameterGroup
修改参数模板	parameterGroup	updateParameterGroup
删除参数模板	parameterGroup	deleteParameterGroup
复制参数模板	parameterGroup	copyParameterGroup
重置参数模板	parameterGroup	resetParameterGroup
应用参数模板	parameterGroup	applyParameterGroup
保存参数模板	parameterGroup	saveParameterGroup
冻结删除	all	rdsUnsubscribeInstance
实例冻结	all	rdsfreezeInstance

5.15.2 查看追踪事件

操作场景

在您开通了云审计服务后，系统开始记录云服务资源的操作。云审计服务管理控制台保存最近7天的操作记录。

本节介绍如何在管理控制台查看最近7天的操作记录。

操作步骤

步骤1 登录管理控制台。

步骤2 选择“管理与监管 > 云审计服务 CTS”，进入云审计服务信息页面。

步骤3 单击左侧导航树的“事件列表”，进入事件列表信息页面。

步骤4 事件列表支持通过筛选来查询对应的操作事件。详细信息如下：

- 事件来源、资源类型和筛选类型：在下拉框中选择查询条件。
其中筛选类型选择资源ID时，还需选择或者手动输入某个具体的资源ID。
- 操作用户：在下拉框中选择某一具体的操作用户。
- 事件级别：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- 时间范围：可通过选择时间段查询操作事件。

步骤5 选择查询条件后，单击“查询”。

步骤6 在需要查看的记录左侧，单击∨展开该记录的详细信息。

步骤7 在需要查看的记录右侧，单击“查看事件”，在弹出框中显示该操作事件结构的详细信息。

步骤8 单击右侧的“导出”，将查询结果以CSV格式的文件导出，该CSV文件包含了云审计服务记录的七天以内的操作事件的所有信息。

关于事件结构的关键字段详解，请参见《云审计服务用户指南》的“事件结构”和“事件样例”章节。

----结束

5.16 日志管理

5.16.1 查看或下载错误日志

关系型数据库服务的日志管理功能支持查看数据库级别的日志，包括数据库主库和从库运行的错误信息，以及运行较慢的SQL查询语句，有助于您分析系统中存在的问题。

查看日志明细

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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。
- 步骤5** 在左侧导航栏单击“日志管理”，在“错误日志”页签下，选择“日志明细”，查看错误日志的详细信息。
- 您可单击页面右上角的级别筛选框查看不同级别的日志记录。

说明

云数据库MySQL实例支持查看以下级别的错误日志：

- ERROR
- WARNING
- NOTE

- 您还可单击右上角的  选择时间区域，查看不同时间段内的错误日志。
- 对于无法完全显示的“描述”，鼠标悬停查看完整信息。

----结束

下载错误日志

- 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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。
- 步骤5** 在左侧导航栏单击“日志管理”，在“错误日志”页签下，选择“下载”页签，对状态为“准备完成”的日志文件，单击操作列中的“下载”，下载错误日志。
- 系统会自动加载下载准备任务，加载时长受日志文件大小及网络环境影响。
 - 下载准备过程中，日志文件状态显示为“准备中...”。
 - 下载准备完成，日志文件状态显示为“准备完成”。
 - 下载准备工作失败，日志文件状态显示为“异常”。“准备中...”和“异常”状态的日志文件不支持下载。
 - 下载链接有效期为5分钟。如果超时，提示用户下载链接已失效，是否重新下载。若需重新下载，单击“确定”，否则单击“取消”。

----结束

5.16.2 查看或下载慢日志

操作场景

慢日志用来记录执行时间超过当前慢日志阈值“long_query_time”（默认是1秒）的语句，您可以通过慢日志的日志明细、统计分析情况，查找出执行效率低的语句，进行优化。您也可以下载慢日志进行业务分析。

关系型数据库服务支持以下执行语句类型：

- SELECT
- INSERT
- UPDATE
- DELETE
- CREATE

查看日志明细

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“日志管理”，在“慢日志”页签下，选择“日志明细”，查看慢SQL语句的详细信息。

- 慢日志功能支持查看指定执行语句类型或时间段的慢日志记录。
- 针对当前的慢日志功能，阈值参数“long_query_time”可以控制SQL响应时间为多少而生成一条慢日志记录，只会影响新增的部分。比如慢日志阈值参数为1s时，上报了超过1s的慢日志记录，后续调整为0.1s，原有上报的日志仍然会展示。

---结束

统计分析慢日志

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“日志管理”，在“慢日志”页签下选择“统计分析”，查看详细信息。

说明

- 慢日志的“统计分析”页签中显示的SQL语句只是同一模式的SQL语句中的一个，作为示例，并不会列出所有的SQL语句。例如：先后执行了语句“select sleep(1)”和“select sleep(2)”，则它们同属于模式“select sleep(N)”，慢日志的结果中只会显示SQL语句“select sleep(1)”。
- 针对当前的慢日志功能，阈值参数“long_query_time”可以控制SQL响应时间为多少而生成一条慢日志记录，只会影响新增的部分。比如慢日志阈值参数为1s时，上报了超过1s的慢日志记录，后续调整为0.1s，原有上报的日志仍然会展示。

---结束

下载慢日志

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“日志管理”，在“慢日志”页签下，选择“下载”页签，对状态为“准备完成”的日志文件，单击操作列中的“下载”，下载慢日志。

- 系统会自动加载下载准备任务，加载时长受日志文件大小及网络环境影响。
 - 下载准备过程中，日志文件状态显示为“准备中...”。
 - 下载准备完成，日志文件状态显示为“准备完成”。
 - 下载准备工作失败，日志文件状态显示为“异常”。“准备中...”和“异常”状态的日志文件不支持下载。
- 下载链接有效期为5分钟。如果超时，提示用户下载链接已失效，是否重新下载。若需重新下载，单击“确定”，否则单击“取消”。

----结束

5.16.3 查看主备切换日志

RDS for MySQL提供主备切换日志。您可以通过MySQL实例的主备切换日志信息，评估对业务的影响。

说明

主备切换日志暂时只支持查询30天之内的日志信息。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“日志管理”，在“主备切换日志”页签下，查看详细信息。

----结束

5.16.4 开启 SQL 审计日志

当您开通SQL审计功能，系统会将所有的SQL操作记录下来存入日志文件，方便用户[下载](#)并查询。

RDS for MySQL默认关闭SQL审计功能，打开可能会有一定的性能影响，本章节指导您如何打开、修改或关闭SQL审计日志。

📖 说明

- 当您开通SQL审计功能，系统会将所有的SQL操作记录下来，并半小时或累积到100M上传存入日志。
- 开启SQL审计，日志文件会占用您的备份空间。

约束限制

仅如下版本支持SQL审计功能。

- MySQL云盘实例支持5.6.43及其以上版本，MySQL本地盘实例支持5.6.47.3及其以上版本
- MySQL云盘实例支持5.7.23及其以上版本，MySQL本地盘实例支持5.7.29.3及其以上版本
- MySQL 8.0版本

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“SQL审计”，单击“设置SQL审计”，在弹出框中设置SQL审计日志保留策略，单击“确定”，保存设置策略。

开启或修改SQL审计：

- 将单击  设置为 ，开启SQL审计日志保留策略。
- 保留天数默认为7天，可设置范围为1~732天。

关闭SQL审计：

将  设置为单击 ，关闭该SQL审计。

- 勾选“确认关闭审计日志后，所有审计日志文件将立即删除。”复选框，表示同意删除审计日志。

须知

SQL审计关闭后，所有审计日志都会被立即删除，不可恢复，请您谨慎操作。

- 不勾选“确认关闭审计日志后，所有审计日志文件将立即删除。”复选框，SQL审计关闭前的审计日志会被继续保留。

----结束

5.16.5 下载 SQL 审计日志

当您**开启SQL审计功能**，系统会将所有的SQL操作记录下来存入日志文件，方便用户进行下载、查询操作。RDS for MySQL默认关闭SQL审计功能，打开可能会有一定的性能影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏单击“SQL审计”。

步骤6 在审计日志列表右上方选择时间范围，勾选目标日志，并单击列表左上方的“下载”，批量下载SQL审计日志。

您还可以选择单个审计日志，单击操作列中的“下载”，下载目标SQL审计日志。

步骤7 SQL审计日志内容如下图所示。字段说明请参考[表5-11](#)。

图 5-5 MySQL 审计日志

```
"6","408543","1159","Connect",2020-03-20T03:35:05 UTC,"","","","","","",
"7","408543","0","Quit",2020-03-20T03:35:05 UTC,"","","","","","",
"8","408544","1159","Connect",2020-03-20T03:35:20 UTC,"","","","","",
"9","408544","0","Quit",2020-03-20T03:35:20 UTC,"","","","","",
"10","408546","1159","Connect",2020-03-20T03:35:35 UTC,"","","","","",
"11","408546","0","Quit",2020-03-20T03:35:35 UTC,"","","","","",
"12","408547","1159","Connect",2020-03-20T03:35:50 UTC,"","","","","",
"13","408547","0","Quit",2020-03-20T03:35:50 UTC,"","","","","",
```

表 5-11 审计日志字段说明

参数	说明
record_id	审计日志单条记录的记录ID。
connection_id	该条记录执行的会话ID，与show processlist中的ID一致。
connection_status	会话状态，常见为执行语句的错误返回码，普通执行成功返回0。
name	记录类型名称，通常情况下dml，ddl操作均为QUERY，连接断开为CONNECT和QUIT。
timestamp	记录的UTC时间。
command_class	执行的SQL命令类型，内部为解析得到的SQL类型，例如select，update（连接断开不存在该项）。
sqltext	执行的SQL具体内容（连接断开审计不存在该项）。

参数	说明
user	登录的账户。
host	登录的host，当本地登录时为localhost，远程登录为空。
external_user	代理用户名称。
ip	通过远程连接的客户端IP，本地连接为空。
default_db	执行SQL时默认的数据库。

----结束

5.17 任务中心

5.17.1 查看任务

您可以通过“任务中心”查看用户在控制台上提交的异步任务的执行进度和状态。目前，RDS支持查看和管理“即时任务”和“定时任务”。

说明

关系型数据库服务支持查看和管理以下任务：

- 创建实例。
- 创建只读实例。
- 磁盘扩容。
- 单实例转主备。
- 实例主备倒换。
- 重启实例。
- 实例绑定弹性IP。
- 实例解绑弹性IP。
- 恢复到新实例。
- MySQL实例迁移可用区。

查看即时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“即时任务”页签下，选择目标任务，查看任务信息。

- 通过任务名称、实例名称/ID确定目标任务，或通过右上角的搜索框输入任务名称来确定目标任务。
- 支持查看某一段时间内的任务执行进度和状态，默认时长为7天。任务保留时长最多为30天。

- 系统支持查看以下状态的即时任务：
 - 执行中
 - 完成
 - 失败
- 查看任务创建时间和结束时间。

----结束

查看定时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“定时任务”页签下，查看目标任务的执行进度和结果。

- 您可以通过实例名称/ID确定目标任务，或通过右上角的搜索框输入实例ID来确定目标任务。
- 系统支持查看以下状态的定时任务：
 - 执行中
 - 完成
 - 失败
 - 取消
 - 待执行
 - 待授权

----结束

5.17.2 删除任务

对于不再需要展示的任务，您可以通过“任务中心”进行任务记录的删除。删除任务仅删除记录，不会删除数据库实例或者停止正在执行中的任务。

须知

删除任务将无法恢复，请谨慎操作。

删除即时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“即时任务”页签下，选择目标任务，单击操作列的“删除”，在弹出框中单击“是”，删除任务。

关系型数据库服务支持删除以下状态的即时任务：

- 执行中
- 完成
- 失败

----结束

删除定时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“定时任务”页签下，选择目标任务，查看目标任务的任务状态，是否是“待执行”或“待授权”状态。

- 是，请执行[步骤5](#)。
- 否，请执行[步骤6](#)。

步骤5 单击操作列的“取消”，在弹出框中单击“是”，取消任务。再次单击操作列的“删除”，在弹出框中单击“是”，删除任务。

步骤6 单击操作列的“删除”，在弹出框中单击“是”，删除任务。

关系型数据库服务支持删除以下状态的定时任务：

- 执行中
- 完成
- 失败
- 取消
- 待执行
- 待授权

----结束

6 PostgreSQL 用户指南

6.1 数据迁移

6.1.1 使用 psql 命令迁移 PostgreSQL 数据

迁移准备

PostgreSQL支持逻辑备份。您可使用pg_dump逻辑备份功能，导出备份文件，再通过psql导入到RDS中，实现将PostgreSQL的数据导入到关系型数据库中。

关系型数据库服务支持开启公网访问功能，通过弹性公网IP进行访问。您也可通过弹性云服务器的内网访问关系型数据库。

准备工作

1. 准备弹性云服务器或可公网访问关系型数据库。
 - 通过弹性云服务器连接关系型数据库实例，需要创建一台弹性云服务器。
 - 通过公网地址连接关系型数据库实例，需具备以下条件。
 - i. 先对关系型数据库实例绑定公网地址，如何绑定公网地址，请参见[绑定弹性公网IP](#)。
 - ii. 保证本地设备可以访问关系型数据库绑定的公网地址。
2. 在1中的弹性云服务器或可访问关系型数据库的设备上，安装PostgreSQL客户端。

说明

该弹性云服务器或可访问关系型数据库的设备上需要安装和RDS PostgreSQL数据库服务端相同版本的数据库客户端，PostgreSQL数据库或客户端会自带pg_dump和psql工具。

导出数据

要将已有的PostgreSQL数据库迁移到关系型数据库，需要先对它进行导出。

须知

- 相应导出工具需要与数据库引擎版本匹配。
- 数据库迁移为离线迁移，您需要停止使用源数据库的应用程序。

步骤1 登录已准备的弹性云服务器，或可访问关系型数据库的设备。

步骤2 使用pg_dump将源数据库导出至SQL文件。

```
pg_dump --username=<DB_USER> --host=<DB_ADDRESS> --port=<DB_PORT> --  
format=plain --file=<BACKUP_FILE> <DB_NAME>
```

- DB_USER为数据库用户。
- DB_ADDRESS为数据库地址。
- DB_PORT为数据库端口。
- BACKUP_FILE为要导出的文件名称。
- DB_NAME为要迁移的数据库名称。

根据命令提示输入数据库密码。

示例如下：

```
$ pg_dump --username=root --host=192.168.151.18 --port=5432 --  
format=plain --file=backup.sql my_db
```

Password for user root:

命令执行完会生成“backup.sql”文件，如下：

```
[rds@localhost ~]$ ll backup.sql
```

```
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 backup.sql
```

----结束

导入数据

步骤1 登录已准备的弹性云服务器，或可访问关系型数据库的设备。

步骤2 确保导入的目标数据库已存在。

如果不存在，执行以下命令创建数据库：

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --  
dbname=postgres -c "create database <DB_NAME>;"
```

- RDS_ADDRESS为RDS实例的IP地址。
- DB_PORT为当前数据库实例的端口。
- DB_NAME为要导入的数据库名称。

步骤3 将导出的文件导入到关系型数据库。

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --  
dbname=<DB_NAME> --file=<BACKUP_DIR>/backup.sql
```

- RDS_ADDRESS为关系型数据库实例的IP地址。

- DB_PORT为当前数据库实例的端口。
- DB_NAME为要导入的目标数据库名称，请确保该数据库已存在。
- BACKUP_DIR为“backup.sql”所在目录。

根据命令提示输入关系型数据库实例的密码。

示例如下：

```
# psql --host=172.16.66.198 --port=5432 --username=root --dbname=my_db --file=backup.sql
```

Password for user root:

步骤4 查看迁移结果，如下。

```
my_db=> \l my_db
```

示例中，名为my_db的数据库已经被导入了：

```
my_db=> \l my_db
List of databases
Name | Owner | Encoding | Collate | Ctype | Access privileges
-----+-----+-----+-----+-----+-----
my_db | root | UTF8 | en_US.UTF-8 | en_US.UTF-8 |
(1 row)
```

----结束

6.2 参数调优

6.2.1 PostgreSQL 参数调优建议

数据库参数是数据库系统运行的关键配置信息，设置不合适的参数值可能会影响业务。本文列举了一些重要参数说明，更多参数详细说明，请参见[PostgreSQL官网](#)。

通过控制台界面修改PostgreSQL参数值，请参见[修改实例参数](#)。

修改敏感参数

若干参数相关说明如下：

- 参数“search_path”值，必须配置为以逗号分隔的模式名序列，并确保该模式名存在，否则会影响数据库的使用。
- 开启参数“log_duration”，可能会在日志中记录带有敏感信息的SQL语句，建议关闭该配置。
- 参数“log_min_duration_statement”用于设置最小执行时间，执行时间大于等于这个值的语句都将被记录，单位为毫秒。值为0，表示记录所有语句，值为-1，表示不记录语句。
- 参数“nls_timestamp_format”和“nls_date_format”可控制timestamp和date数据类型输入输出格式，仅支持PostgreSQL增强版。
 - “nls_timestamp_format”是时间戳数据类型格式，“nls_date_format”是日期数据类型格式，两者更改其一即可生效。
- 参数“temp_file_limit”是每个会话可使用的所有临时文件的最大值，单位是KB，仅支持PostgreSQL 11和PostgreSQL 12。该操作是高危操作，请确认后配置。

- 如果配置的参数值超过约束时，会导致实例异常。
- 如果将参数值修改为较大值供临时使用，使用完成后没有将参数修改为原来的值，可能会导致误用临时空间导致占满磁盘停止用户业务，且用户实例变为异常。

修改性能参数

若干参数相关说明如下：

- 参数“log_statement”配置为“ddl”、“mod”或“all”，会记录创建及删除数据库用户的操作，包含数据库用户的密码等敏感信息，同时也会影响数据库的性能，请确认后配置。
- 开启参数“log_hostname”、“log_duration”、“log_connections”和“log_disconnections”会影响数据库的性能，请确认后配置。

6.3 PostgreSQL 增强版

6.3.1 PostgreSQL 增强版简介

PostgreSQL增强版是基于开源PostgreSQL 11数据库，自主研发的数据库服务，完整的保留了PostgreSQL数据库的所有功能。

范围涵盖：SYS系统视图、PL/SQL、数据类型、高级函数、SQL语法、Null值处理等。

本产品为企业客户上云降低成本，提供高安全、高可用、高性能的整体解决方案。

6.3.2 函数

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的内置函数和高级函数包。

表 6-1 内置函数

内置函数	说明
add_months(date,integer)	返回date加上integer的值，返回值为DATE类型。
appendchildxml(XMLType_instance, XPath_string, value_expr[, namespace_string])	在XMLType类型数据XMLType_instance指定位置XPath_string处追加value_expr节点，其中namespace_string用于描述XPath_string的命名空间信息。
asciistr(string)	返回给定字符串string对应的ASCII值，不支持非ASCII字符。
bin_to_num(expr_list)	将expr_list中的二进制串转换为对应的十进制数，返回值类型为NUMBER。
bitand(number1,number2)	返回number1和number2按位与的值，返回值为BIT类型。

内置函数	说明
convert(char, dest_char_set[, source_char_set])	将source_char_set字符集编码格式的输入字符串char, 转换为dest_char_set字符集编码格式, 该函数只在服务端生效。
cosh(n)	返回参数n的双曲余弦值;
decode(expr,search1, result1 [[,search2, result2],.....] [, default])	将expr表达式值依次与所有search表达式 (search1, search2.....) 进行比较, 如果与searchn进行匹配, 则返回resultn, 否则返回default。如果没有default, 返回null。
empty_blob()	返回一个空BLOB类型。
hexraw(char)	将十六进制字符串转换为RAW类型的值。
instrb(string, substring[, position[, occurrence]])	返回从字符串string的第position位置开始, 第occurrence次出现的子串substring起始位置, 默认从字符串string首部开始第一次出现的位置。
last_day(date)	返回date所在月的最后一天。
lengthb(char)	返回char字符的字节长度, char支持的类型为所有字符串类型 (如CHAR, VARCHAR2, NCHAR, NVARCHAR2等) 或可隐式转换为字符串的类型 (如integer等) 。
listagg(measure_expr[, 'delimiter']) within group(order_by_clause) [over query_partition_clause]	将query_partition_clause分组中的每个列表表达式measure_expr值按order_by_clause规则排序后, 再合并成一行, 值之间使用delimiter分隔符进行分隔。
lnvl(condition)	返回条件表达式condition的取反值, 返回类型为BOOLEAN。
mod(n2, n1)	支持除数为0的情况, 即当n1为0时, 直接返回n2。
months_between(date1, date2)	返回date1与date2之间相差的月数, 当date2早于date1, 返回值为负数。
nanvl(n2, n1)	当单精度或双精度浮点数n2的值为NAN, 则返回n1, 否则返回n2。
nchr(number)	返回本地字符集中二进制与number相等的字符。
new_time(date, timezone1, timezone2)	返回timezone1的date时间对应的timezone2的时间, 返回值类型为DATE。
next_day(date, char)	返回date后的第一个自然日 (包括工作日、休息日和节假日), 返回值为DATE类型。

内置函数	说明
numtodsinterval(n, interval_unit)	根据INTERVAL数据类型单元interval_unit（取值范围：'DAY'、'HOUR'、'MINUTE'、'SECOND'），将数字n转换为数据类型INTERVAL DAY TO SECOND。
numtoyminterval(n, 'interval_unit')	将数字n按照指定单位interval_unit转为数据类型INTERVAL YEAR TO MONTH，其中interval_unit取值为：YEAR、MONTH。
nlsort(char[, nlsparam])	按照nlsparam指定的排序字符集对字符串char进行排序，默认使用char字符串字符集排序；
nls_upper(char[, nlsparam])	根据nlsparam指定的排序规则将字符串char中的字母转换为大写，其中char字符串类型为CHAR, VARCHAR2, NCHAR, NVARCHAR2,CLOB或NCLOB，nlsparam的取值为'NLS_SORT = sort'。
nls_lower(char[, nlsparam])	根据nlsparam指定的排序规则将字符串char中的字母转换为小写，其中char字符串类型为CHAR, VARCHAR2, NCHAR, NVARCHAR2,CLOB或NCLOB，nlsparam的取值为'NLS_SORT = sort'。
nvl(expr1, expr2)	返回expr1, expr2中第一个非空值。
rawtohex(raw)	将RAW类型的值转换为十六进制字符串。
regexp_count(source_char, pattern, position, match_param)	返回指定字符串source_char中的指定位置position开始，匹配正则表达式模式pattern的次数。其中，match_param参数会影响正则表达式匹配规则，比如match_param='i'忽略大小写等。
regexp_instr(source_char, pattern[, position[, occurrence[, return_opt[, match_param[, subexpr]]]])	该函数拓展了INSTR函数的功能，允许使用正则表达式匹配，返回值类型为INTEGER。 <ul style="list-style-type: none"> • position表示查找起始位置。 • occurrence表示查找pattern在source_char的第几次出现。 • return_opt: <ul style="list-style-type: none"> - 取值为0表示返回模式匹配的起始位置。 - 取值为1表示返回模式匹配的结束位置。 • match_param表示正则表达式模式匹配控制参数，如区分大小写等。 • subexpr表示正则表达式分组匹配的组号。
regexp_like(source_char, pattern[, match_param])	source_char为字符串表达式，pattern为正则表达式，match_param为正则表达式控制参数，返回字符串source_char是否可按照正则表达式pattern进行匹配。

内置函数	说明
regexp_substr(source_char, pattern[,position[,occurrence[,match_param[,subexpr]]]])	按正则表达式在source_char字符串中匹配子字符串。 <ul style="list-style-type: none"> source_char为查找的输入字符串，支持所有字符串类型（如CHAR, VARCHAR2, NCHAR, NVARCHAR2等）或可隐式转换为字符串的类型（如integer等）。 pattern为子字符串匹配的正则表达式。 position为指定匹配的起始字符位置。 occurrence为pattern在source_char出现的次数。 match_parameter为正则表达式控制参数。 subexpr为pattern的第几个子表达式，范围为0~9。
raise_application_error(errnum, errmsg)	将错误码errnum与错误消息errmsg输出到客户端。
remainder(n2, n1)	返回n2/n1的余数，类似于mod，区别在于，mod取余时用了floor处理，而remainder使用round处理。返回值类型为NUMERIC或双精度浮点数（由入参类型决定）。
round(n,precision)	返回n的四舍五入值，precision为精度值。
scn_to_timestamp(number)	返回SCN号number产生的最近时间戳；
sinh(n)	返回数字n的双曲正弦值，当n类型为BINARY_FLOAT，返回类型BINARY_DOUBLE，否则返回值类型为NUMERIC。
substr(char,position[,substring_length])	返回char字符串中第position个字符开始，长度为substring_length的子字符串。若不指定substring_length，则截取到字符串结尾。
substrb(char, position[, substring_length])	返回char字符串中第position个字节开始，长度为substring_length字节的子字符串。若不指定substring_length，则截取到字符串结尾。
sys_context(namespace, parameter)	返回指定参数parameter在命名空间namespace下的值，返回值类型为VARCHAR2。
sys_guid()	返回RAW类型的全局唯一标识。
sys_connect_by_path(column, char)	仅适用于CONNECT BY查询，返回column列的根节点；
tanh(n)	返回参数n的双曲正切值；
to_blob(char)	将char字符串转换为BLOB类型，char支持的类型为所有字符串类型（如CHAR, VARCHAR2, NCHAR, NVARCHAR2等）或可隐式转换为字符串的类型（如integer等）。

内置函数	说明
to_binary_float(expr)	将数字字符串expr转换为单精度float类型。
to_binary_double(expr)	将数字字符串expr转换为双精度float类型。
to_clob(char)	将char字符串转换为CLOB数据类型。
to_char(char)	增加char支持的类型: char、character、varchar。
to_date(char[,fmt])	将char时间字符串按照fmt格式转换为date数据类型, char支持的类型有CHAR、VARCHAR2、NCHAR、NVARCHAR2、TIMESTAMP。如果省略fmt, 则char必须采用DATE数据类型的默认格式。
to_dsinterval('sql_format' 'ds_iso_format')	将SQL标准(如'100 00:00:00')或ISO标准(如'P100DT05H')的时间字符串转换为数据类型INTERVAL DAY TO SECOND。
to_multi_byte(char)	将单字节字符char转换成对应的多字节字符。
to_number(expr)	将expr表达式值转换为number类型。
to_number(expr, fmt, 'nlsparam')	按照指定格式fmt转换expr字符串为数字, 返回类型为NUMBER, 其中nlsparam为国际化语言参数, 支持的参数分别为: NLS_NUMERIC_CHARACTERS、NLS_CURRENCY、NLS_ISO_CURRENCY。
to_timestamp(char[,fmt])	将char时间字符串按照fmt格式转换为timestamp数据类型, char支持的类型有CHAR、VARCHAR2、NCHAR、NVARCHAR2、TIMESTAMP。如果省略fmt, 则char必须采用TIMESTAMP数据类型的默认格式。
to_single_byte(char)	将多字节输入字符串转换为单字节字符串。
to_ymininterval('sql_format' 'ym_iso_format')	将SQL标准(如'01-02')或ISO标准(如'P1Y2M')的时间字符串转换为数据类型INTERVAL MONTH TO YEAR。
timestamp_to_scn(timestamp)	根据时间戳timestamp返回系统变更号SCN;
trunc(date[, fmt])	按照fmt指定的日期格式对date进行截断处理, 返回值类型为DATE。如果省略fmt, 则默认日期格式为'DDD'。
tz_offset({time_zone_name '{+ -}hh:mi'})	返回指定时区的偏移量, 返回值类型VARCHAR2, 参数可以是时区名time_zone_name或者'{+ -}hh:mi'格式字符串。
value(correlation_variable)	以对象表的方式返回correlation_variable所关联表的记录行, 返回类型为correlation_variable所关联的对象表。

表 6-2 高级函数包

高级函数包	说明
DBMS_OUTPUT.PUT(item)	将item字符串放入本地缓冲区；item为所有可以转换为字符串的类型。
DBMS_OUTPUT.PUT_LINE(item)	将item字符串放入本地缓冲区，然后将整个本地缓冲区的内容整体输出；item为所有可以转换为字符串的类型。
DBMS_RANDOM.SEED(val)	val为生成随机数的种子，可为字符串和数字类型。
DBMS_RANDOM.VALUE([low,high])	返回low和high之间长度为16位的随机数，如果不指定low和high的范围，则默认范围为0~1。
dbms_lob.getlength(lob_loc {clob blob})	返回lob_loc指定的LOB对象长度；
dbms_lob.read(lob_loc, amount, offset, buffer)	从指定偏移offset读取LOB对象lob_loc指定长度amount的内容写入缓冲区buffer；
dbms_lob.write(lob_loc, amount, offset, buffer)	将buffer缓冲器的内容写入大对象lob_loc缓冲区（并不影响所引用的大对象），从offset开始写入长度为amount。
utl_raw.cast_to_raw(char)	将VARCHAR2类型字符串char转换为RAW类型，返回值类型为RAW。
utl_raw.length(raw)	返回raw数据类型的字节长度，返回值类型为NUMBER。
utl_raw.cast_from_binary_integer(n, endianness)	按照endianness指定的内存对齐方式将整数n转换为RAW类型，其中endianness取值分别为： <ul style="list-style-type: none"> • 1，大端对齐； • 2，小端对齐； • 3，按机器对齐方式；

6.3.3 系统视图

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的系统视图。

表 6-3 系统视图

超级管理员	DBA	USER
ALL_ALL_TABLES	DBA_ALL_TABLES	-
ALL_COL_COMMENTS	-	USER_COL_COMMENTS

超级管理员	DBA	USER
-	DBA_DATA_FILES	-
ALL_DIRECTORIES	DBA_DIRECTORIES	-
ALL_INDEXES	DBA_INDEXES	USER_INDEXES
ALL_JOBS	DBA_JOBS	USER_JOBS
ALL_OBJECTS	-	USER_OBJECTS
ALL_PROCEDURES	DBA_PROCEDURES	USER_PROCEDURES
ALL_SOURCE	DBA_SOURCE	USER_SOURCE
ALL_SEQUENCES	DBA_SEQUENCES	USER_SEQUENCES
ALL_TABLES	DBA_TABLES	USER_TABLES
-	DBA_TABLESPACES	USER_TABLESPACE
ALL_TAB_COLUMNS	DBA_TAB_COLUMNS	USER_TAB_COLUMNS
-	DBA_TRIGGERS	USER_TRIGGERS
ALL_USERS	DBA_USERS	-
ALL_VIEWS	DBA_VIEWS	USER_VIEWS
ALL_IND_COLUMNS	DBA_IND_COLUMNS	USER_IND_COLUMNS
ALL_TAB_PARTITIONS	DBA_TAB_PARTITIONS	USER_TAB_PARTITIONS
ALL_PART_TABLES	DBA_PART_TABLES	USER_PART_TABLES
ALL_PART_KEY_COLUMNS	DBA_PART_KEY_COLUMNS	USER_PART_KEY_COLUMNS
ALL_PART_INDEXES	DBA_PART_INDEXES	USER_PART_INDEXES
ALL_TAB_SUBPARTITIONS	DBA_TAB_SUBPARTITIONS	USER_TAB_SUBPARTITIONS
ALL_SUBPART_KEY_COLUMNS	DBA_SUBPART_KEY_COLUMNS	USER_SUBPART_KEY_COLUMNS

表 6-4 普通视图

视图名称	说明
V\$SESSION	展示了与当前会话相关的信息，如SID，username等。
NLS_SESSION_PARAMETERS	展示了当前会话的NLS参数及取值。
V\$SESSION_LONGOPS	显示运行时间超过6秒的数据库操作的状态。

6.3.4 数据类型

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的数据类型。

表 6-5 数据类型

数据类型名称	数据类型
变长字符串类型	VARCHAR2, NVARCHAR2
十进制浮点数类型	DECIMAL
双精度二进制浮点类型	BINARY_DOUBLE
二进制数据类型	RAW
二进制大对象类型	BLOB
字符大对象类型	CLOB
字节字符大对象类型	NCLOB
数字类型	NUMBER
变长字符串类型	NVARACHAR
Unicode字符数据类型	NCHAR
三十二位浮点型数据类型	BINARY_FLOAT
长整型	LONG
XML数据类型	XMLType
本地时间戳数据类型	TIMESTAMP WITH LOCAL TIME ZONE
PL/SQL 整型溢出翻转数据类型	BINARY_INTEGER
PL/SQL 整型溢出错误数据类型	PLS_INTEGER

6.3.5 隐式类型转换

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的隐式类型转换。

- 定长字符串类型CHARACTER与NUMERIC, INT4, INT8, FLOAT4, FLOAT8之间相互转换。
- 变长字符串类型VARCHAR与NUMERIC, INT4, INT8, FLOAT4, FLOAT8之间相互转换。
- 文本类型TEXT与NUMERIC, INT2, INT4, INT8, FLOAT4, FLOAT8之间相互转换。

- 短整形INT2转换为CHARACTER, VARCHAR。
- 二进制大对象BLOB与二进制RAW之间相互转换。

6.3.6 预定义参数

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的预定义参数。

表 6-6 预定义参数

预定义参数	说明
NLS_DATE_FORMAT	日期格式定义参数。
NLS_DATE_LANGUAGE	日期语言定义参数。
NLS_DUAL_CURRENCY	区域名称定义参数，影响本地货币符号显示格式；
NLS_CURRENCY	货币符号定义参数。
NLS_TIME_FORMAT	不带时区的时间格式定义参数。
NLS_TIME_TZ_FORMAT	带时区的时间格式定义参数。
NLS_TIMESTAMP_FORMAT	不带时区的时间戳格式定义参数。
NLS_TIMESTAMP_TZ_FORMAT	带时区的时间戳格式定义参数。
NLS_NUMERIC_CHARACTERS	十进制数分组分割符定义参数。
NLS_ISO_CURRENCY	区域名称定义参数，影响ISO货币符号显示格式。
NLS_TERRITORY	区域名称定义参数，会根据区域货币与数字显示格式重置NLS_CURRENCY、NLS_ISO_CURRENCY、NLS_NUMERIC_CHARACTERS参数值。
NLS_LANGUAGE	区域语言定义参数。
NLS_LENGTH_SEMANTICS	指定字符串字段定义时的默认长度单位，取值范围：BYTE, CHAR；
NLS_SORT	指定本地字符集排序规则。
NLS_COMP	指定会话级范围查找或排序操作的排序规则。

6.3.7 宏变量

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的宏变量。

- SYSDATE：获取当前系统时期。
- SYSTIMESTAMP：获取当前系统时间戳。

- DBTIMEZONE：获取当前数据库时区。
- SESSIONTIMEZONE：获取当前会话时区。
- ROWNUM：获取查询结果中的元组序号。

6.3.8 操作符

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的操作符。

- 求差集操作符：MINUS
- 不等于操作符：^=

说明

操作符^= <> != >= <=中间允许出现空白字符（包括空格、Tab键等）。

6.3.9 语法

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版支持的语法。

- 支持CREATE SEQUENCE语法。
- 支持CREATE/ALTER DATABASE语法。
- 支持CREATE/ALTER VIEW语法。
- 支持CREATE TABLE语法。
- 支持CREATE TABLESPACE语法。
- 支持CLUSTER语法。
- 支持FORALL语法；
- 支持CREATE/DROP DIRECTORY语法。
- 支持ALTER TABLE ADD CONSTRAINT USING INDEX语法；
- 支持INSERT INTO语句的目标列名可以使用表名或表别名修饰。
- 支持非分区表的ROWNUM伪列；
- 支持字段表达式创建索引：CREATE INDEX ON COLUMN_EXPR；
- 支持ALTER TABLE...MODIFY修改表字段。
- 支持VARCHAR、CHARACTER数据类型指定长度单位。
- 支持TYPE/NAME/VERSION/VALUE/INTERVAL作为别名。
- 支持存储过程语法。
- 支持DATE数据类型。
- 支持HASH、RANGE、LIST分区表创建语法。
- 支持MERGE操作语法：
MERGE [HINT] INTO table_name USING ({subquery | table_name | view_name}) alias ON
(condition) merge_update_clause merge_insert_clause;
- 支持时间间隔操作语法：
INTERVAL YEAR TO MONTH,INTERVAL DAY (I) TO SECOND (P);
- 支持CREATE TRIGGER带BODY体语法：
CREATE TRIGGER name... {DECLARE ... BEGIN | BEGIN} body END;
- 支持存储过程游标语法：
CURSOR cursor_name [parameter_list] IS select_statement, TYPE type_name IS REF
CURSOR;

- 支持存储过程游标变量：
SQL%ISOPEN,SQL%FOUND,SQL%NOTFOUND,SQL%ROWCOUNT,cursor%ISOPEN,cursor%FOUND,cursor%NOTFOUND,cursor%ROWCOUNT;
- 支持定时任务高级包：
DBMS_JOB.SUBMIT,DBMS_JOB.ISUBMIT,DBMS_JOB.REMOVE,DBMS_JOB.BROKEN,DBMS_JOB.CHANGE,DBMS_JOB.WHAT,DBMS_JOB.NEXT_DATE,DBMS_JOB.INTERVAL;
- 支持CREATE USER语法：
{DEFAULT COLLATION | DEFAULT TABLESPACE | [LOCAL] TEMPORARY TABLESPACE} Clause;
- 支持修改会话属性：
ALTER SESSION SET param_name = value;
- 支持匿名块。
- 支持存储过程跨模式访问。
- 存储过程支持SQLCODE内置变量；
- 存储过程语法兼容增强：支持以存储过程名作为结束标签，支持FOR VAR IN SELECT-CLAUSE，LOOP循环允许指定结束标签名，支持in参数指定默认值；
- 支持子查询不指定别名。
- CREATE SEQUENCE支持NOCYCLE。
- CREATE/ALTER USER语法中允许使用IDENTIFIED BY关键字替代PASSWORD关键字。
- UPDATE SET语法中允许指定表名或别名修饰。
- UPDATE SET支持单字段(columnname)=(value)语法。
- ALTER TABLE支持MODIFY NOT NULL语法和ENABLE语法。
- 空字符串和NULL等价。
- 序列操作新增语法：sequencCURRVAL，sequencNEXTVAL。
- 创建用户时支持同时创建同名schema。
- 删除表记录语法中FROM关键字。
- 支持XML数据类型伪列COLUMN_VALUE。
- 支持外连接运算符(+)；
- 支持数据类型INTERVAL与数字之间的运算操作：+、-、>、<、>=、<=、<>；
- 支持分区表的DML操作：SELECT、INSERT、UPDATE、DELETE；
- 分区表支持二级分区；
- 分区表支持以表达式作为分区边界；
- 支持触发器DDL：支持使用schema修饰等；
- 支持时间格式IYY；
- 支持CREATE/ALTER MATERIALIZED VIEW语法；
- 支持CREATE TYPE语法；
- 支持create profile语法；
- 支持列约束的enable/disable语法；
- 支持分区表指定tablespace options语法；
- 支持DROP TABLE tablename [CASCADE CONSTRAINTS] [PURGE]语法；
- 支持存储过程动态SQL语法EXECUTE IMMEDIATE，当前版本不支持带DECLARE声明的匿名块动态执行；

- 支持FUNCTION定义；
- 支持CONNECT BY查询：支持伪列LEVEL、CONNECT_BY_ROOT、CONNECT_BY_ISLEAF；支持函数sys_connect_by_path；支持CONNECT_BY_ROOT操作；支持ORDER SIBLINGS；
- 支持TIME数据类型精度；
- 支持虚拟列:column_name datatype [GENERATED ALWAYS] AS (expression) [VIRTUAL]；
- 支持一维数组的定义：CREATE OR REPLACE TYPE array_name AS VARRAY (len) OF typename；
- 支持一维数组的访问：array_name.extend, array_name.count, array_name.first, array_name.last；
- ROLLUP、CUBE、GROUPING SETS分组查询支持grouping_id([expr1[, expr2[, ...exprn]])和group_id()；
- 支持无分组字段返回的排序查询语句：SELECT SUM(colname) FROM tbl ORDER BY colname；

6.3.10 功能增强

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版新增内置插件PG_PERMISSIONS以及新增的分区表优化和查询并行功能优化。

分区表优化

- 支持按hash key分区。
- 支持分区表PRIMARY KEY， FOREIGN KEY， INDEXES， TRIGGERS。
- 允许创建“default”分区。
- 支持INSERT ON CONFLICT。
- 支持自动移动UPDATE所影响的记录到正确的分区。
- 支持智能并行JOIN（需要打开enable_partitionwise_join开关）。

查询并行优化

PostgreSQL增强版提高了并行查询性能，同时并行顺序扫描和散列连接的性能得到提升，对分区数据的扫描也更高效。如果底层查询无法并行化，PostgreSQL现在可以执行并行使用UNION的SELECT查询。

- 创建B-tree索引支持并行处理。
- CREATE TABLE ... AS， CREATE MATERIALIZED VIEW以及部分查询支持并行UNION。
- 支持hash并行JOIN。

6.3.11 安全加固

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版新增安全加固内容。

- 日志中禁止打印密码等敏感信息。
- 用户组权限管理优化：仅允许组内成员授权该用户组的权限。

- 客户端与服务端密码认证加密提供SHA256加密。
- 解决安全漏洞CVE-2018-16850。
- 优化pg_stat_statements统计信息查看权限控制。

6.3.12 其他功能

本章节介绍了在PostgreSQL11开源版本的基础上，PostgreSQL增强版新增的其他功能。

- 支持伪表DUAL。
`SELECT * FROM DUAL;`
- 新增配置参数“sql_format”，该参数默认值为“postgresql”。
- 优化CREATE TABLESPACE云应用场景路径管理，用户对真实路径无感知。
- 引入pg_hint_plan插件，增强查询计划优化维护手段。
- 提供数据文件完整性校验工具pg_verify_checksums。
- 支持定时任务管理功能。
- 增强pg_stat_statements插件功能。

6.4 PostgreSQL 内核功能

6.4.1 RDS for PostgreSQL 内核版本发布记录

RDS for PostgreSQL支持的社区版本的发布日期如下表所示。

表 6-7 RDS for PostgreSQL 支持的社区版本

发布日期	支持的社区版本
2022-08	<ul style="list-style-type: none">• 14.4• 13.7• 12.11• 11.16• 10.21
2022-05	<ul style="list-style-type: none">• 13.6• 12.10• 11.15• 10.20• 9.6.24

发布日期	支持的社区版本
2021-05	<ul style="list-style-type: none">● 13.2● 12.6● 11.11● 10.16● 9.5.25
2020-05	<ul style="list-style-type: none">● 12.2● 11.5● 10.10● 9.6.15● 9.5.19

6.5 实例生命周期

6.5.1 创建与原实例相同配置的数据库实例

操作场景

关系型数据库服务支持用户快速创建相同配置的数据库实例。

说明

- 创建相同配置实例没有次数限制。
- 只读实例不能执行创建相同配置的操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 创建相同配置”。

步骤5 在“创建相同配置”页面，各项配置默认和原实例信息相同，您可根据业务需要进行修改，填选完毕后，单击“立即创建”。

PostgreSQL实例配置相关信息，请参见[步骤一：创建实例](#)。

步骤6 确认实例信息。

步骤7 稍后刷新实例列表，查看创建结果。如果实例状态为“正常”，说明实例创建成功。

创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

----结束

6.5.2 重启实例

操作场景

通常出于维护目的，您可能需要重启数据库实例。例如：对于某些运行参数修改，需要重启单个实例使之生效。您可通过控制台对主实例和只读实例执行重启操作。

约束限制

- 如果数据库实例处于“异常”状态，可能会重启失败。
- 重启数据库实例会重新启动数据库引擎服务。重启数据库实例将导致短暂中断，在此期间，数据库实例状态将显示为“重启中”。
- 重启过程中，实例将不可用。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主实例，或者单击  展开只读实例，单击“更多 > 重启实例”。

您也可以在“实例管理”页面单击目标实例名称，进入“基本信息”页面。在页面右上角，单击“重启实例”。

重启实例时，如果是主备实例，对应的备实例也会被同步重启。

步骤5 在“重启实例”弹框，选择重启时间，单击“是”重启实例。

步骤6 稍后刷新实例列表，查看重启结果。如果实例状态为“正常”，说明实例重启成功。

----结束

6.5.3 自定义列表项

操作场景

您可以根据自身业务需要，自定义设置实例信息列表项。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击 ，选择自定义列表项。

- 系统默认展示以下列表项：实例名称/ID、实例类型、数据库引擎版本、运行状态、内网地址、操作。

默认实例信息列表项不可更改。

- 单project情况下，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。
- 多project情况下，需要同时开通项目管理服务权限，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。

----结束

6.5.4 导出实例

操作场景

您可以导出所有实例，或根据一定条件筛选出来的目标实例，查看并分析实例信息。

导出所有实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例列表右上角 ，默认导出所有的数据库实例，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

导出筛选的目标实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，根据引擎类型、实例名称、实例ID、实例内网地址等条件，筛选实例，或勾选需要导出的实例，单击实例列表右上角 ，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

6.5.5 删除按需实例

操作场景

您可根据业务需要，在关系型数据库“实例管理”页面手动删除实例来释放资源。

约束限制

- 正在执行操作的实例不能手动删除，只有在实例操作完成后，才可删除实例。
- 实例删除后将不再产生费用，实例生成的自动备份会被同步删除，保留的手动备份会继续收取费用。

须知

- 删除主实例时，会同步删除其对应的备实例和只读实例，请您谨慎操作。
- 实例删除后，资源立即释放，请谨慎操作。如需保留数据，请务必确认完成[数据备份](#)后再删除实例。
- 通过保留的手动备份，可以恢复实例数据，具体请参见[通过备份文件恢复关系型数据库实例数据](#)。

删除主实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择需要删除的主实例，在“操作”列，选择“更多 > 删除实例”。

步骤5 在“删除实例”弹框，单击“是”下发请求。

步骤6 稍后刷新“实例管理”页面，查看删除结果。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择对应主实例，单击实例名称前的 ，可查看到该主实例下的全部只读实例。

步骤5 选择目标只读实例，在“操作”列，选择“更多 > 删除实例”。

步骤6 在“删除实例”弹框，单击“是”下发请求。

步骤7 稍后刷新“实例管理”页面，查看删除结果。

----结束

6.5.6 回收站

操作场景

RDS支持将删除的主备或者单机实例，加入回收站管理。通过在回收站中重建实例，将数据恢复到新实例上。新实例数据库引擎、数据库版本、存储类型与原实例相同，其他参数可以重新配置。默认可以恢复1~7天内删除的实例。

约束限制

- RDS不回收只读实例，只有主备或者单机实例才会进入回收站。
- 回收站策略机制默认开启，且不可关闭。

设置回收站策略

须知

修改回收站保留天数，仅对修改后新进入回收站的实例生效，对于修改前已经存在的实例，仍保持原来的回收策略，请您谨慎操作。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在左侧导航栏，单击“回收站”。

步骤5 在“回收站”页面，单击“回收站策略”，设置已删除实例保留天数，可设置范围为1~7天。

步骤6 单击“确定”，完成设置。

----结束

重建实例

在回收站保留期限内的主实例可以通过重建实例恢复数据。

步骤1 在左侧导航栏，单击“回收站”。

步骤2 在“回收站”页面，在实例列表中找到需要恢复的目标实例，单击操作列的“重建”。

步骤3 在“重建新实例”页面，选填配置后，提交重建任务，具体可参考[恢复到新实例操作](#)。

----结束

6.6 变更实例

6.6.1 修改实例名称

操作场景

关系型数据库服务支持修改主实例和只读实例的实例名称，以方便用户识别。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称后的 ，编辑实例名称，单击“确认”，即可修改实例名称。

您也可以单击目标实例名称，进入实例的“基本信息”页面，在“数据库信息”模块实例名称处，单击 ，修改实例名称。

实例名称长度4~64个字符，必须以字母开头，可包含大写字母、小写字母、数字、中划线、下划线或中文，不能包含其他特殊字符。

- 单击 ，提交修改。
- 单击 ，取消修改。

步骤5 在实例的“基本信息”页面，查看修改结果。

----结束

6.6.2 切换主备实例的可用性策略

操作场景

关系型数据库服务支持切换主备实例的可用性策略，以满足不同业务需求。可选择“可靠性优先”或者“可用性优先”两种策略。

- 可靠性优先（默认选择）：数据库应该尽可能保障数据的可靠性，即数据丢失量最少。对于数据一致性要求较高的业务，建议选择该策略。
- 可用性优先：数据库应该尽可能快速恢复服务，即可用时间最长。对于数据库在线时间要求较高的业务，建议选择该策略。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击主实例名称。

步骤5 在“基本信息”页面，在“数据库信息”模块的“切换策略”处，单击“修改”，在弹出框中，选择策略，单击“确定”，完成修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

6.6.3 变更实例的 CPU 和内存规格

操作场景

CPU/内存规格可根据业务需要进行变更，当实例的状态由“规格变更中”变为“正常”，则说明变更成功。

约束限制

- 当实例进行CPU/内存规格变更时，该实例不可被删除。
- 如果主实例下存在只读实例，主实例规格变更时，所选变更规格需要小于等于只读实例的规格；同样只读实例规格变更时，所选规格需要大于等于主实例当前的规格。
- 修改CPU/内存后，将会重启数据库实例。请选择业务低峰期，避免业务异常中断。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 规格变更”，进入“规格变更”页面。

您也可以通过单击目标实例名称，进入“基本信息”页面，在“数据库信息”模块的“性能规格”处，单击“规格变更”，进入“规格变更”页面。

步骤5 在“规格变更”页面，选择所需修改的性能规格，单击“下一步”。

步骤6 查看变更结果。

任务提交成功后，单击“返回云数据库RDS列表”，在实例管理页面，可以看到实例状态为“规格变更中”。稍后在对应的“基本信息”页面，查看实例规格，检查修改是否成功。此过程需要5~15分钟。

须知

RDS PostgreSQL实例变更成功后，系统将根据新内存大小，调整如下参数的值：

- “shared_buffers”
- “max_connections”
- “maintenance_work_mem”
- “effective_cache_size”

----结束

6.6.4 磁盘扩容

操作场景

随着业务数据的增加，原来申请的数据库磁盘容量可能会不足，需要为关系型数据库实例进行扩容。

当实例处于“磁盘空间满”状态时，需扩容至磁盘空间使用率小于85%才可使实例处于可用状态，使数据库恢复正常的写入操作。建议您设置“磁盘使用率”指标的告警规则，及时了解磁盘使用情况，起到预警作用，具体请参见[设置告警规则](#)。

数据库磁盘空间不足的原因和解决方案，请参见[数据超过了RDS实例的最大存储容量怎么办](#)。

磁盘扩容期间，服务不中断。关系型数据库服务只支持扩容磁盘存储空间大小，不支持变更存储类型。

约束限制

- 云数据库PostgreSQL实例最大可扩容至4000GB，扩容次数没有限制。
- 如果是主备实例，针对主实例扩容时，会同时对其备实例进行扩容。
- 扩容过程中，该实例不可重启和删除。
- 磁盘容量变更只允许扩容，不能缩容。

主实例扩容

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 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。

您还可以通过如下途径进行扩容：

- 单击目标实例名称，进入“基本信息”页面。在“存储空间”模块，单击“磁盘扩容”，进入“磁盘扩容”页面。

步骤5 在“服务选型”页面，选择空间大小，“下一步”。

每次扩容最小容量为10GB，实例所选容量大小必须为10的整数倍，最大磁盘容量为4000GB。

步骤6 规格确认。

- 重新选择：单击“上一步”，回到上个页面，修改新增大小。
- 确认无误：单击“提交申请”，提交扩容。
- 确认无误：单击“提交”，提交扩容。

步骤7 查看扩容结果。

在实例管理页面，可看到实例状态为“扩容中”，稍后单击实例名称，在“基本信息”页面，查看磁盘大小，检查扩容是否成功。此过程需要3~5分钟。

----结束

只读实例扩容

只读实例磁盘扩容与主实例的磁盘扩容互不影响，因此，您需要单独为只读实例扩容，以满足业务需求。对只读实例进行扩容时，您可选择大于或等于主实例的存储空间。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择相应的实例，单击实例名称前的∨可查看到只读实例，单击“操作”列的“更多 > 磁盘扩容”。

您还可以通过如下途径进行扩容：

- 单击目标实例名称，进入“基本信息”页面。在“存储空间”模块，单击“磁盘扩容”，进入“磁盘扩容”页面。

步骤5 在“服务选型”页面，选择空间大小，单击“立即申请”。

每次扩容最小容量为10GB，实例所选容量大小必须为10的整数倍，最大磁盘容量为4000GB。

步骤6 规格确认。

- 重新选择：单击“上一步”，回到上个页面，修改新增大小。
- 确认无误：单击“提交申请”，提交扩容。
- 确认无误：单击“提交”，提交扩容。

步骤7 查看扩容结果。

在实例管理页面，可看到实例状态为“扩容中”，稍后单击实例名称，在“基本信息”页面，查看磁盘大小，检查扩容是否成功。此过程需要3~5分钟。

----结束

6.6.5 单机转主备实例

操作场景

- 关系型数据库支持数据库单机实例转为主备实例。在保留原实例资源的情况下提高了实例的可靠性。
- 主备实例可实现自动故障倒换，备机快速接管业务。建议您将主备实例选择到不同的可用区，享受跨可用区，同城容灾的高可用服务。
- 支持主备实例的反亲和性部署，避免单个物理机故障导致整个实例不可用。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择单机实例，单击“操作”列的“更多 > 转主备”，进入“转主备”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例左侧的添加按钮



，进入“转主备”页面。

步骤5 在转主备时，您只需选择“备可用区”，其他信息默认与主实例相同。确认信息无误，单击“立即申请”。

步骤6 单机转主备创建成功后，单击“返回云数据库RDS列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 创建过程中，状态显示为“转主备中”。您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。

- 在实例列表的右上角，单击  刷新列表，可查看到单机转主备完成后，实例状态显示为“正常”，“实例类型”显示为“主备”。

----结束

6.6.6 手动切换主备实例

操作场景

当关系型数据库主备类型的实例创建成功后，系统会在同一区域内为该实例创建一个备实例。主备实例的数据会实时同步，用户仅能访问主实例数据，备实例仅作为备份形式存在。您可根据业务需要，进行主备实例的切换，切换后原来的主实例会变成备实例，可实现机架级别的容灾。

约束限制

- 实例运行正常。
- 主备实例复制关系正常。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在“基本信息”页面中“数据库信息”模块的“实例类型”处，单击“主备切换”。

您也可以在“基本信息”页面，单击“实例拓扑图”模块的。进行主备切换。

须知

主备切换可能会造成几秒或几分钟的服务闪断（闪断时间与复制时延有关），并有可能在主备同步时延过大的情况下，导致少量数据丢失。请在业务低峰期进行主备切换，避免对高峰期业务造成影响。

步骤6 在“主备切换”弹框，单击“是”进行主备实例的切换。

在“复制状态”为“正常”的情况下，如果复制时延大于300s，主备切换任务无法下发。

步骤7 主备切换成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行检查和管理。

- 切换过程中，状态显示为“主备切换中”。
- 在实例列表的右上角，单击刷新列表，可查看到主备切换完成后，实例状态显示为“正常”。

----结束

6.6.7 迁移可用区

操作场景

您可以将主备实例的备机迁移至同一区域内的其它可用区。

说明

- 迁移期间将短暂停止DDL语句和event定时任务，请选择业务低峰期操作，避免业务异常中断。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 可用区迁移”，进入“可用区迁移”页面。

步骤5 在“可用区迁移”页面，选择目标可用区，单击“提交”。

步骤6 迁移可用区成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行检查和管理。

- 可用区迁移过程中，状态显示为“备机迁移中”。您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。
- 在实例列表的右上角，单击刷新列表，可查看到可用区迁移完成后，实例状态显示为“正常”。

- 在“基本信息”页面中“数据库信息”模块的“可用区”处，可以查看到备机迁移后所在的可用区。

----结束

6.7 只读实例

6.7.1 只读实例简介

产品简介

目前，云数据库PostgreSQL的实例支持只读实例。

在对数据库有少量写请求，但有大量读请求的应用场景下，单个实例可能无法抵抗读取压力，甚至对主业务产生影响。为了实现读取能力的弹性扩展，分担数据库压力，您可以在某个区域中创建一个或多个只读实例，利用只读实例满足大量的数据库读取需求，以此增加应用的吞吐量。您需要在应用程序中分别配置主实例和每个只读实例的连接地址，才能实现将写请求发往主实例而将读请求发往只读实例。

只读实例为单个节点的架构（没有备节点），采用PostgreSQL的原生复制功能将主实例的更改同步到所有只读实例，而且主实例和只读实例之间的数据同步不受网络延时的影响，只读实例跟主实例在同一区域，但可以在不同的可用区。

功能特点

- 只读实例规格可以与主实例不一致，建议只读实例规格大于等于主实例规格，否则易导致只读实例延迟高、负载高等现象。
- 不需要维护账号与数据库，全部通过主实例同步。
- 提供系统性能监控。关系型数据库提供近20个系统性能的监控视图，如磁盘容量、IOPS、连接数、CPU利用率、网络流量等，用户可以轻松查看实例的负载。
- 备份设置：不支持自动备份设置以及手动备份。
- 实例恢复：不支持通过备份文件或任意时间点创建只读实例，不支持通过备份集覆盖只读实例。
- 数据迁移：不支持将数据迁移至只读实例。
- 数据库管理：不支持创建和删除数据库。
- 账号管理：只读实例不提供创建账号权限。
- 性能规格设置：创建只读实例所选规格，必须大于等于当前主实例的规格。

功能限制

- 1个主实例最多可以创建5个只读实例。

创建和管理只读实例

- [创建只读实例](#)
- [管理只读实例](#)

6.7.2 创建只读实例

操作场景

只读实例用于增强主实例的读能力，减轻主实例负载。

关系型数据库主实例或主备实例创建成功后，可根据业务需要创建只读实例。

说明

一个主实例中，最多可以增加5个只读实例。

创建只读实例的规格需要大于等于当前主实例的规格。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击操作列的“创建只读”，进入“服务选型”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例下方的添加按钮



，创建只读实例。

步骤5 在“服务选型”页面，填选实例相关信息后，单击“立即创建”。

表 6-8 基本信息

参数	描述
当前区域	只读实例默认与主实例在同一区域。
实例名称	实例名称的长度在4~64个字符之间，必须以字母开头，可包含大写字母、小写字母、数字、中划线、下划线或中文，不能包含其他特殊字符。
数据库引擎	默认与主实例的数据库引擎一致，不可更改。
数据库版本	默认与主实例的数据库版本一致，不可更改。
可用区	关系型数据库服务支持在同一个可用区内或者跨可用区部署数据库主实例和只读实例，以提高可靠性。

表 6-9 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。 关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。 只读实例的存储空间大小默认与主实例一致。

表 6-10 网络

参数	描述
虚拟私有云	和主实例相同。
子网	和主实例相同，创建只读实例时RDS会自动为您配置内网地址，您也可输入子网号段内未使用的内网地址，实例创建成功后该内网地址可修改。
安全组	和主实例相同。

步骤6 在“规格确认”页面，进行信息确认。

- 如果需要重新选择，单击“上一步”，回到服务选型页面修改基本信息。
- 信息确认无误，单击“提交”，下发新增只读实例请求。

步骤7 只读实例创建成功后，用户可以在“实例管理”页面，选择只读实例所对应的实例，单击对其进行查看和管理。

您也可以在基本信息页面的“实例拓扑图”中，单击只读实例的名称，进入该只读实例的“基本信息”页面，对其进行查看和管理。

----结束

后续操作

[管理只读实例](#)

6.7.3 管理只读实例

通过只读实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的 ，单击目标只读实例的名称，进入实例的基本信息页面，即进入只读实例的管理页面。

----结束

通过主实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 单击只读实例所在的主实例名称，进入主实例的“基本信息”页面。

步骤5 在“实例拓扑图”中单击只读实例的名称，进入该只读实例的“基本信息”页面，即进入只读实例的管理页面。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的 ，选择目标只读实例，单击操作列中的“更多 > 删除实例”。

----结束

6.8 备份与恢复

6.8.1 备份概述

关系型数据库支持数据库实例的备份和恢复，以保证数据可靠性。

自动备份

关系型数据库服务会在数据库实例的备份时段中创建数据库实例的自动备份。系统根据您指定的备份保留期保存数据库实例的自动备份。如果需要，您可以将数据恢复到备份保留期中的任意时间点。具体请参见[设置自动备份策略](#)。

手动备份

您还可以创建手动备份对数据库进行备份，手动备份是由用户启动的数据库实例的全量备份，手动备份会一直保存，直到用户手动删除。具体请参见[创建手动备份](#)。

下载备份文件

您可以下载全量备份和增量备份文件，用于本地存储备份或者恢复数据库。具体请参见[下载全量备份文件](#)和[下载增量备份文件](#)。

6.8.2 设置自动备份策略

操作场景

创建关系型数据库实例时，系统默认开启自动备份策略，安全考虑，实例创建成功后不可关闭，您可根据业务需要设置自动备份策略，关系型数据库服务按照您设置的自动备份策略对数据库进行备份。

关系型数据库服务的备份操作是实例级的，而不是数据库级的。当数据库故障或数据损坏时，可以通过备份恢复数据库，从而保证数据可靠性。备份以压缩包的形式存储在对象存储服务上，以保证用户数据的机密性和持久性。由于开启备份会损耗数据库读写性能，建议您选择业务低峰时间段设置自动备份。

关系型数据库默认开启的自动备份策略设置如下：

- 保留天数：默认为7天。超过保留天数的备份文件将会被删除且无法恢复。
- 备份时间段：默认为24小时中，间隔一小时的随机的一个时间段，例如01:00 ~ 02:00，12:00 ~ 13:00等。备份时间段以UTC时区保存。如果碰到夏令时/冬令时切换，备份时间段会因时区变化而改变。
- 备份周期：默认为一周内的每一天。

查看或修改自动备份策略

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏，单击“备份恢复”，单击“修改备份策略”。您可以查看到已设置的备份策略，如需修改备份策略，请调整以下参数的值。

- 保留天数是指自动备份可保留的时间，增加保留天数可提升数据可靠性，请根据需要设置。
- 减少保留天数的情况下，该备份策略对已有备份文件同时生效，即超出备份保留天数的已有备份文件会被删除。

- 保留天数为全量自动备份和增量备份的保留时长，范围为1~732天，备份时间段为间隔1小时，建议根据业务情况，选择业务低峰时段，备份周期默认全选，可修改，且至少选择一周中的一天。

步骤6 单击“确定”，确认修改。

----结束

6.8.3 设置跨区域备份策略

操作场景

关系型数据库支持将备份文件存放到另一个区域存储，某一区域的实例故障后，可以在异地区域使用备份文件在异地恢复到新的RDS实例，用来恢复业务。

实例开启跨区域备份策略后，会自动将该实例的备份文件备份到目标区域。您可以在RDS“备份管理”页面，对跨区域的备份文件进行管理。

开启或修改跨区域备份策略

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏，单击“备份恢复”，单击“设置跨区域备份策略”。

- 开启跨区域全备：将本实例的自动全量备份文件备份到异地。
- 开启跨区域日志备份：将本实例的全量备份和增量备份文件备份到异地。
- 跨区域备份文件的保留天数范围为1~1825天。
- 开启跨区域备份策略仅对新产生的备份生效，即为开启跨区域备份策略后产生的备份生效，之前的备份文件不会存储到异地。
- 开启跨区域日志备份后需要待下一次自动全量备份复制完成后才可进行将数据库实例恢复到指定时间点操作，并且只允许恢复到自动全量备份完成后的时间点。
- 同一个租户下各数据库实例，备份选择的目标端区域需要相同。
- 跨区域备份会在实例本区域备份完成后异步复制到异地。

步骤6 单击“确定”，确认修改。

步骤7 您可以在“备份管理”页面的“数据库跨区域备份”页签，管理跨区域备份文件。

- 单击“设置区域备份”，可以修改跨区域备份策略。
- 单击“查看跨区域备份”，可以查看到生成的备份文件，当原区域实例发生故障，可以通过该备份文件将数据恢复到新实例。

----结束

关闭跨区域备份策略

步骤1 在“备份管理”页面，选择“数据库跨区域备份”。

步骤2 在目标实例上，单击“设置区域备份”。在弹出页面，通过设置开关，关闭跨区域备份策略。

步骤3 单击“确定”，确认关闭。

----结束

6.8.4 创建手动备份

操作场景

关系型数据库服务支持对运行正常的主实例创建手动备份，用户可以通过手动备份恢复数据，从而保证数据可靠性。

说明

当数据库实例被删除时，关系型数据库实例的自动备份将被同步删除，手动备份不会被删除。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，在操作列选择“更多 > 创建备份”。

步骤5 在创建备份弹出框中，命名该备份，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!<"&'=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏，单击“备份恢复”，对其进行查看并管理。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“备份恢复”，单击“创建备份”，命名该备份，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!"&'=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏中选择“备份恢复”，对其进行查看并管理。

----结束

6.8.5 下载全量备份文件

操作场景

用户可以下载手动和自动备份文件，用于本地存储备份或者恢复数据库。

云数据库PostgreSQL支持用户下载全量备份文件。

方式 1：直接浏览器下载

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要下载的可用备份，单击操作列中的“下载”。

您也可进入目标实例的“基本信息”页面，在左侧导航栏选择“备份恢复”，在“全量备份”页签下，单击操作列中的“下载”。

步骤5 在弹出框中，单击“直接下载”，单击“确定”，通过浏览器直接下载数据库实例的备份文件。

步骤6 您可根据业务需要，在本地进行数据恢复。

----结束

方式 2：按地址下载

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要下载的可用备份，单击操作列中的“下载”。

您也可进入目标实例的“基本信息”页面，在左侧导航栏选择“备份恢复”，在“全量备份”页签下，单击操作列中的“下载”。

步骤5 在弹出框中，单击“按地址下载”，单击  复制地址，通过下载地址下载备份文件。

对于PostgreSQL，显示当前备份文件在链接有效期内的下载地址。

- 您可以使用其他下载工具下载备份文件。
- 您还可以使用 `wget` 命令下载备份文件：
`wget -O FILE_NAME --no-check-certificate "DOWNLOAD_URL"`

命令中的参数解释如下：

`FILE_NAME`：下载成功后的备份文件名称，由于原始文件名称较长，可能会超出客户端文件系统的限制，建议下载备份文件时使用“-O”进行重命名。

`DOWNLOAD_URL`：需下载的备份文件所在路径，如果包含特殊字符则需要转义。

步骤6 您可根据业务需要，在本地进行数据恢复。

----结束

6.8.6 下载增量备份文件

操作场景

用户可以下载手动和自动备份文件，用于本地存储备份或者恢复数据库。

云数据库 PostgreSQL 支持用户下载增量备份文件。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称，进入“基本信息”页面，在左侧导航栏，单击“备份恢复”，在“增量备份”页签下，单击操作列中的“下载”。

您也可以勾选需要下载的增量备份，单击左上角“下载”。

步骤5 下载任务执行完成后，您可在本地查看到增量备份文件。

----结束

6.8.7 通过备份文件恢复关系型数据库实例数据

操作场景

关系型数据库服务支持使用已有的自动备份和手动备份，将实例数据恢复到备份被创建时的状态。该操作恢复的为整个实例的数据。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在左侧导航栏单击“备份管理”，选择需要恢复的备份，单击操作列的“恢复”。

您也可在“实例管理”页面，单击指定的实例名称，在左侧导航栏单击“备份恢复”，在“全量备份”页签下单击目标备份对应的操作列中的“恢复”。

步骤5 选择需要的恢复方式，单击“确定”。

- 新实例：将备份恢复到新创建的实例上。
跳转到“恢复到新实例”的服务选型页面：
 - 数据库引擎和数据库版本，与原实例相同，不可重置。
 - 存储空间大小默认和原实例相同，且必须大于或等于原实例存储空间大小。
 - 其他参数，用户如需设置，请参见[步骤一：创建实例](#)。
- 已有实例：将备份数据恢复到已经存在的实例上。
 - a. 勾选“我确认恢复到目标实例会导致数据被覆盖，恢复过程中数据库不可用。目标实例引擎类型、版本号要求与原实例相同，且存储空间大于等于原实例。”，单击“下一步”。
 - b. 确认恢复信息无误，单击“确定”。

须知

- 如果目标数据库实例已被删除，则不可恢复到已有实例。
- 恢复到已有实例会导致实例数据被覆盖，且恢复过程中实例将不可用。
- 只可选择与原实例相同VPC，相同引擎，相同版本或高版本的实例。
- 请确保目标实例的存储空间大于或等于当前实例的存储空间，否则会导致任务下发失败。

步骤6 查看恢复结果。

- 恢复到新实例
RDS会为用户重新创建一个和该备份数据相同的实例。可看到实例由“创建中”变为“正常”，说明恢复成功。
恢复成功的新实例是一个独立的实例，与原有实例没有关联。如需使用只读实例，请重新在该实例上进行创建。
新实例创建成功后，系统会自动执行一次全量备份。
- 恢复到已有实例
在“实例管理”页面，可查看目标实例状态为“恢复中”，恢复完成后，实例状态由“恢复中”变为“正常”。如果目标实例下存在只读实例，只读实例的状态与目标实例一致。
恢复成功后，会执行一次全量备份。

----结束

6.8.8 将数据库实例恢复到指定时间点

操作场景

关系型数据库服务支持使用已有的自动备份，恢复实例数据到指定时间点。

实例恢复到指定时间点，会从OBS备份空间中选择一个该时间点最近的全量备份下载到实例上进行全量恢复，再重放增量备份到指定时间点，恢复时长和实例的数据量有关，平均恢复速率为30MB/s。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“备份恢复”页签，单击“恢复到指定时间点”。

步骤6 选择恢复日期和该日期下可恢复的时间区间，并输入要恢复到的时间点，选择恢复方式，单击“确定”。

- 恢复到新实例

跳转到“恢复到新实例”的服务选型页面：

- 数据库引擎和数据库版本，与原实例相同，不可修改。
- 存储空间大小默认和原实例相同，且必须大于或等于原实例存储空间大小。
- 其他参数默认，用户可设置，请参见[步骤一：创建实例](#)。

- 恢复到已有实例

- a. 勾选“我确认恢复到目标实例会导致数据被覆盖，恢复过程中数据库不可用。目标实例引擎类型、版本号要求与原实例相同，且存储空间大于等于原实例。”，单击“下一步”。
- b. 确认恢复信息无误，单击“确定”。

须知

- 恢复到已有实例会导致实例数据被覆盖，且恢复过程中实例将不可用。
 - 只可选择与原实例相同VPC，相同引擎，相同版本或高版本的实例。
 - 请确保目标实例的存储空间大于或等于当前实例的存储空间，否则会导致任务下发失败。
-

步骤7 查看恢复结果。

- 恢复到新实例

RDS会为用户重新创建一个和该时间点数据相同的实例。可看到实例由“创建中”变为“正常”，说明恢复成功。

恢复成功的新实例是一个独立的实例，与原有实例没有关联。如需使用只读实例，请重新在该实例上进行创建。

创建或恢复完成后，系统会自动执行一次全量备份。

- 恢复到已有实例

在“实例管理”页面，可查看该实例下所有实例状态为“恢复中”，恢复完成后，实例状态由“恢复中”变为“正常”。

创建或恢复完成后，系统会自动执行一次全量备份。

----结束

6.8.9 复制备份

操作场景

关系型数据库服务支持复制自动和手动备份，制作一个不同名的拷贝。

限制条件

只支持在同一区域下复制并使用备份，暂不支持跨region操作。

备份保留策略

- 如果需要长期保留自动备份，您可以复制该自动备份生成一个手动备份，该手动备份在您删除之前将会一直保留。
- 如果手动备份所使用的空间超过您的默认存储空间，则可能会增加关系型数据库存储成本。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入“基本信息”页面，在左侧导航栏，单击“备份恢复”，选择目标备份，单击操作列中的“复制”。

步骤5 填写新备份名称和描述，单击“确定”。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，不能包含回车和>!<"&'=特殊字符。

步骤6 新备份创建成功后，用户可在“备份管理”页面，查看并管理自己的备份。

----结束

6.9 参数模板管理

6.9.1 创建参数模板

您可以使用数据库参数模板中的参数来管理数据库引擎配置。数据库参数模板就像是引擎配置值的容器，这些值可应用于一个或多个数据库实例。

默认参数模板包含数据库引擎默认值和系统默认值，具体根据引擎、计算等级及实例的分配存储空间而定。您无法修改默认数据库参数模板的参数设置，您必须创建自己的数据库参数模板才能更改参数设置的默认值。

须知

并非所有数据库引擎参数都可在客户创建的数据库参数模板中进行更改。

如果您想使用您自己的数据库参数模板，只需创建一个新的数据库参数模板，创建实例的时候选择该参数模板，如果是在创建实例后有这个需求，可以重新应用该参数模板，请参见[应用参数模板](#)。

若您已成功创建数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案，请参见[复制参数模板](#)。

以下是您在使用数据库参数模板中的参数时应了解的几个要点：

- 当您更改动态参数并保存数据库参数模板时，将立即应用更改。当您更改静态参数并保存数据库参数模板时，参数更改将在您手动重启数据库实例后生效。
- 在数据库参数模板内设置参数不恰当可能会产生意外的不利影响，包括性能降低和系统不稳定。修改数据库参数时应始终保持谨慎，且修改数据库参数模板前要备份数据。将参数模板更改应用于生产数据库实例前，您应当在测试数据库实例上试用这些参数模板设置更改。

说明

关系型数据库和文档数据库服务不共享参数模板配额。

每个用户最多可以创建100个关系型数据库参数模板，各关系型数据库引擎共享该配额。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，单击“创建参数模板”。

步骤5 选择数据库引擎版本，命名参数模板并添加对该参数模板的描述，单击“确定”，创建参数模板。

- 选择该数据库引擎参数模板所需应用的参数模板类型。
- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<"&'=特殊字符。

---结束

6.9.2 修改实例参数

为确保关系型数据库服务发挥出最优性能，用户可根据业务需求对用户创建的参数模板中的参数进行调整。

您可以修改用户创建的数据库参数模板中的参数值，但不能更改默认数据库参数模板中的参数值。

如果您更改一个参数值，则所做更改的应用时间将由该参数的类型决定。

关系型数据库服务的管理控制台显示与数据库实例关联的数据库参数模板的状态。例如，如果数据库实例未使用与其关联的数据库参数模板所做的最新更改，则关系型数据库服务的管理控制台将显示状态为“等待重启”。您将需要手动重启数据库实例，以使最新的参数更改对该数据库实例生效。

📖 说明

系统提供的默认参数模板不允许修改，只可单击参数模板名进行查看。当用户参数设置不合理导致数据库无法启动时，可参考默认参数模板重新配置。

修改当前实例的参数

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“参数修改”，在“参数”页签修改相应参数。

可进行的操作如下：

须知

根据参数列表中“是否需要重启”提示，进行相应操作：

- 是：在实例列表中，查看“运行状态”，如果显示“参数变更，等待重启”，则需重启实例使之生效。
 - 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
 - 修改只读实例的某些参数，需要重启该只读实例使之生效。
- 否：无需重启，立即生效。

- 单击“保存”，在弹出框中单击“确定”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

参数修改完成后，您可[查看参数修改历史](#)。

---结束

批量修改参数

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，选择需要编辑的自定义参数模板，单击参数模板名称。

步骤5 默认在“参数”页签下，根据需要修改相关参数。

可进行的操作如下：

- 单击“保存”，在弹出框中单击“确定”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

步骤6 参数修改完成后，您可以单击“参数修改历史”查看参数的修改详情。

步骤7 参数模板修改后，不会立即应用到当前使用的实例，您需要应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤8 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

----结束

6.9.3 导出参数模板

操作场景

- 您可以将您创建的数据库实例参数列表，导出并生成一个新的参数模板，供您后期使用。请参考[应用参数模板](#)将导出的参数模板应用到新的实例。
- 您可以将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV中，方便查看并分析。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”页签单击“导出”。

- 导出到参数模板。将该实例对应参数列表导出并生成一个参数模板，供您后期使用。

在弹出框中，填写新参数模板名称和描述，单击“确定”。

说明

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!'<'&'=特殊字符。

创建完成后，会生成一个新的参数模板，您可在“参数模板管理”页面的对其进行管理。

- 导出到文件。将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV表中，方便用户查看并分析。

在弹出框中，填写文件名称，单击“确定”。

📖 说明

文件名称长度在4~64个字符之间，必须以字母开头，可以包含字母、数字、中划线或下划线，不能包含其他特殊字符。

----结束

6.9.4 比较参数模板

操作场景

您可以比较同数据库类型的实例参数模板，以了解当前实例参数的差异项。

您也可以比较同数据库类型的默认参数模板，以了解当前参数模板的配置情况。

比较当前实例参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”子页签中单击“比较参数”。

步骤6 在弹出框中选择当前实例同数据库类型的参数模板，单击“确定”，比较两个参数的差异项。

- 有差异项，则会显示差异参数的如下信息：参数名称、当前实例参数模板的参数值和被比较参数模板的参数值。
- 无差异项，则不显示。

----结束

比较目标参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“比较”。

步骤5 选择同一数据库引擎的不同参数模板，单击“确定”，比较两个参数模板之间的配置参数差异项。

- 有差异项，则会显示差异参数模板的如下信息：参数名称、两个参数模板的参数值。

- 无差异项，则不显示。

----结束

6.9.5 查看参数修改历史

操作场景

您可以查看当前实例所使用参数模板以及自定义参数模板的修改历史，以满足业务需要。

说明

用户创建或导出的新参数模板，在未进行参数修改前，无修改历史。

查看当前实例的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏，单击“参数修改”，单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态、修改时间、是否应用以及应用时间。

如修改后参数模板未应用，请根据业务需要，参考[应用参数模板](#)，将其应用到对应实例。

----结束

查看目标参数模板的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，单击目标参数模板名称。

步骤5 单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态和修改时间。

----结束

6.9.6 复制参数模板

操作场景

您可以复制您创建的自定义数据库参数模板。当您已创建一个数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案。

复制数据库参数模板之后，新参数模板可能不会立即显示，建议您等待5分钟再使用。

您无法复制默认参数模板。不过，您可以创建基于默认参数模板的新参数模板。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要复制的参数模板，单击“复制”。

您还可以在“实例管理”页面，单击实例名称，在左侧导航栏，单击“参数修改”，单击“导出”，将该实例对应参数列表导出并生成一个参数模板，供您后期使用。

步骤5 在弹出框中，填写新参数模板名称和描述，单击“确定”。

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<''&'=特殊字符。

创建完成后，会生成一个新的参数模板，您可在参数模板列表中对其进行管理。

----结束

6.9.7 重置参数模板

操作场景

您可根据自己的业务需求，重置自己创建的参数模板对应的所有参数，使其恢复到默认值。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要设置的参数模板，单击“更多 > 重置”。

步骤5 单击“是”，重置所有参数为其默认值。

步骤6 参数模板修改后，不会立即应用到当前使用的实例，您需要进行应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤7 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

----结束

6.9.8 应用参数模板

操作场景

参数模板编辑修改后，您可以根据业务需要应用到实例中，参数模板只能应用于相同版本的实例中。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，根据参数模板类型不同进行如下操作。

- 若需要将默认参数模板应用到实例，在“系统默认”页签的目标参数模板单击“应用”。
- 若需要将用户自己创建的参数模板应用到实例，在“自定义”页签的目标参数模板单击“更多 > 应用”。

一个参数模板可被应用到一个或多个实例。

步骤5 在弹出框中，选择或输入所需应用的实例，单击“确定”。

参数模板应用成功后，您可[查看参数模板应用记录](#)。

----结束

6.9.9 查看参数模板应用记录

操作场景

参数模板编辑修改后，您可根据业务需要将其应用到对应实例中，RDS支持查看参数模板所应用到实例的记录。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 单击“参数模板管理”。
- 步骤5** 在“系统默认”页签下，选择目标参数模板，单击“应用记录”；或在“自定义”页签下，选择目标参数模板，单击“更多 > 应用记录”，查看应用记录。
- 您可查看参数模板所应用到的实例名称/ID、应用状态、应用时间、失败原因。
- 结束

6.9.10 修改参数模板描述

操作场景

参数模板创建成功后，用户可根据需要对自己创建的参数模板描述进行修改。

说明

默认参数模板的描述不可修改。

操作步骤

- 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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“描述”列 。
- 步骤5** 输入新的描述信息，单击 ，提交修改，单击 ，取消修改。
- 参数模板的描述长度不能超过256个字符，且不能包含>!<"&'=特殊字符。
 - 修改成功后，可在参数模板列表的“描述”列查看改后的描述信息。
- 结束

6.9.11 删除参数模板

操作场景

您可删除废弃的参数模板。

须知

- 参数模板删除后，不可恢复，请谨慎操作。
- 默认参数模板不可被删除。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要删除的参数模板，单击“更多 > 删除”。

步骤5 单击“是”，删除参数模板。

----结束

6.10 连接管理

6.10.1 配置和修改内网地址

操作场景

用户从线下或者其他云迁移到关系型数据库后要面对更改IP的问题，为减少客户业务更改，降低迁移难度。提供规划与更改内网IP方式，降低客户迁移成本。

约束限制

修改内网IP后域名需要几分钟重新解析地址导致数据库连接中断，请在业务停止期间操作。

配置内网 IP

在创建实例时，可在“服务选型”页面的网络部分，根据选择的子网自动配置内网地址。

修改内网 IP

对于创建完成的关系型数据库实例，支持更改内网地址。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 在“基本信息”页“连接信息”模块“内网地址”处，单击“修改”。

步骤6 在“修改内网地址”弹出框中进行修改。单击“确定”，保存修改内容。

已使用IP地址列表中使用情况为“待使用”的IP地址表示已被占用，不允许被当前实例使用。

----结束

6.10.2 绑定和解绑弹性公网 IP

操作场景

须知

为保证数据库可正常访问，请确保数据库使用的安全组开通了相关端口的访问权限，假设数据库的访问端口是5432，那么需确保安全组开通了5432端口的访问。

前提条件

- 只有主实例和只读实例才能绑定弹性公网IP。
- 对于已绑定弹性公网IP的实例，需解绑后，才可重新绑定其他弹性公网IP。

绑定弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 选择“弹性公网IP”页面，单击“绑定弹性IP”。

步骤6 在弹出框的弹性公网IP地址列表中，显示“未绑定”状态的弹性公网IP，选择所需绑定的弹性公网IP，单击“确定”，提交绑定任务。如果没有可用的弹性公网IP，单击“查看弹性IP”，获取弹性公网IP。

步骤7 在“弹性公网IP”页面，查看绑定成功的弹性公网IP。

您也可以在“任务中心”页面，查看绑定弹性公网IP任务的执行进度及结果。

如需关闭，请参见[解绑弹性公网IP](#)。

----结束

解绑弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 对于已绑定弹性公网IP的实例，在“实例管理”页面，选择指定实例，单击实例名称，进入实例基本信息页面。

步骤5 选择“弹性公网IP”页面，单击IP地址后面的“解绑”，在弹出框中单击“是”，解绑弹性公网IP。

步骤6 在“弹性公网IP”页面，查看结果。

您也可以在“任务中心”页面，查看解绑弹性公网IP任务的执行进度及结果。

如需重新绑定，请参见[绑定弹性公网IP](#)。

----结束

6.10.3 修改数据库端口

操作场景

关系型数据库服务支持修改主实例和只读实例的数据库端口，对于主备实例，修改主实例的数据库端口，该实例下备实例的数据库端口会被同步修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击主实例名称，或单击，单击只读实例名称。

步骤5 在“基本信息”页面，在“连接信息”模块“数据库端口”处，单击，修改数据库端口。

说明

PostgreSQL数据库端口修改范围为2100~9500。

- 单击，提交修改。
 - 在弹出框中，单击“是”，提交修改。
 - i. 修改主实例数据库端口，对应的主备实例均会被修改且重启。
 - ii. 修改只读实例数据库端口，仅修改并重启该只读实例。
 - iii. 此过程需要1~5分钟左右。
 - 在弹出框中，单击“取消”，取消本次修改。
- 单击，取消修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

6.10.4 通过 pgAdmin 连接实例

您可通过pgAdmin客户端可视化操作，连接关系型数据库实例。

须知

请使用pgAdmin 4及其以上版本。

准备工作

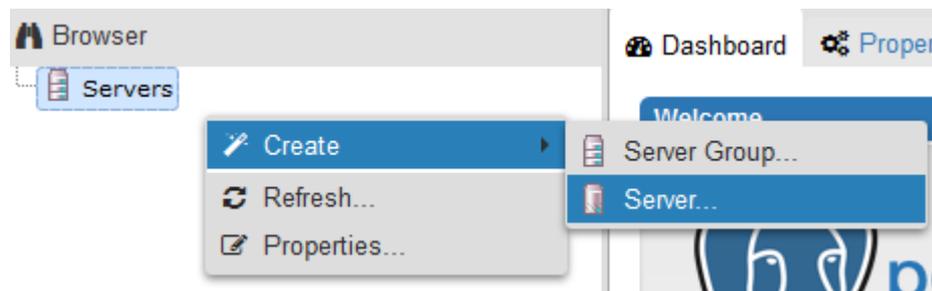
1. 准备**弹性云服务器**或可访问关系型数据库实例的设备。
通过内网地址连接关系型数据库实例，需具备以下条件。
 - 该弹性云服务器与目标实例必须处于同一VPC内。
 - 该弹性云服务器必须处于目标实例所属安全组允许访问的范围内。通过公网地址连接关系型数据库实例，需具备以下条件。
 - a. 保证本地设备可以访问关系型数据库实例绑定的公网地址。
2. 在**1**中的**弹性云服务器**或可访问关系型数据库的设备上，安装pgAdmin客户端。

操作步骤

步骤1 启动pgAdmin客户端。

步骤2 在打开的登录信息窗口，单击并选择“Servers>Create>Server”。

图 6-1 创建



步骤3 在弹出框的“General”页签，输入“Name”，在“Connection”页签，输入要连接的关系型数据库实例信息，单击“Save”。

图 6-2 General

The screenshot shows the 'Create - Server' dialog box with the 'General' tab selected. The 'Name' field is empty and highlighted with a red border. The 'Server group' dropdown is set to 'Servers'. The 'Connect now?' checkbox is checked. The 'Comments' field is empty. A red error bar at the bottom of the dialog contains the text 'Name must be specified.' The bottom of the dialog has three buttons: 'Save', 'Cancel', and 'Reset'.

图 6-3 连接信息

The screenshot shows the 'Create - Server' dialog box with the 'Connection' tab selected. The 'Host name/address' field is empty and highlighted with a red border. The 'Port' field contains '5432'. The 'Maintenance database' field contains 'postgres'. The 'User name' field is empty and highlighted with a red border. The 'Password' field is empty. The 'Save password?' checkbox is unchecked. The 'Role' field is empty. The 'SSL mode' dropdown is set to 'Prefer'. A red error bar at the bottom of the dialog contains the text 'Name must be specified.' The bottom of the dialog has three buttons: 'Save', 'Cancel', and 'Reset'.

关键参数解释：

- Host name/address: 使用内网方式连接时，输入目标实例基本信息页面“连接信息”模块的“内网地址”；使用公网方式连接时，输入目标实例的弹性公网IP地址。
- port: 输入数据库端口，默认5432。
- User name: 默认root。
- Password: 要访问关系型数据库实例的User name所对应的密码。

步骤4 在打开登录信息窗口，若连接信息无误，即会成功连接实例。

----结束

6.11 插件管理

6.11.1 创建和删除插件

提供基于root用户的PostgreSQL插件管理方案，下列插件无需用户手动创建，其他插件均需要参考本章节内容手动创建。

- auto_explain
- passwordcheck
- pg_profile_pro
- pg_sql_history
- wal2json
- test_decoding

说明

PostgreSQL插件是数据库级生效，并不是全局生效。因此创建插件时需要在对应的业务库上都进行手动创建。

PostgreSQL 11、PostgreSQL 增强版和PostgreSQL 12的最新小版本，支持以root用户通过社区的方式来创建(create extension)、删除(drop extension)插件。

创建插件

步骤1 以root用户连接数据库，以database1为例，并使用模板库template1创建需要支持插件的数据库。

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --dbname=database1 --  
username=root -c "create database <DB_NAME> template template1;"
```

- *RDS_ADDRESS*为RDS实例的IP地址。
- *DB_PORT*为RDS数据库实例的端口。
- *DB_NAME*为需要创建插件的数据库名称。

根据提示输入root用户密码。

创建需要支持插件的数据库my_extension_db示例如下：

```
# psql --host=192.168.6.141 --port=5432 --dbname=database1 --  
username=root -c "create database my_extension_db template template1;"
```

```
Password for user root:  
CREATE DATABASE
```

注：如果需要操作的是普通用户user1创建的数据库db1，需要先以普通用户user1登录db1（登录方式参考上述内容），然后执行如下语句将db1权限授予root用户：

```
GRANT ALL ON DATABASE db1 TO root;
```

步骤2 以root用户连接需要支持插件的数据库，并创建插件。

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --dbname=<DB_NAME> --  
username=root -c "select control_extension('create','<EXTENSION_NAME>');"
```

- *RDS_ADDRESS*为RDS实例的IP地址。
- *DB_PORT*为RDS数据库实例的端口。
- *DB_NAME*为需要创建插件的数据库名称。
- *EXTENSION_NAME*为插件名称，请参见[支持的插件列表](#)。

根据提示输入root用户密码。

在数据库my_extension_db中创建postgis插件示例如下：

```
# psql --host=192.168.6.141 --port=5432 --dbname=my_extension_db --  
username=root -c "select control_extension('create','postgis');"
```

```
Password for user root:  
control_extension  
-----  
create postgis successfully.  
(1 row)
```

----结束

删除插件

以root用户连接已创建插件的数据库，并删除插件。

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --  
dbname=<DB_NAME> -c "select control_extension  
( 'drop','<EXTENSION_NAME>');"
```

- *RDS_ADDRESS*为RDS实例的IP地址。
- *DB_PORT*为RDS数据库实例的端口。
- *DB_NAME*为需要创建插件的数据库名称。
- *EXTENSION_NAME*为插件名称，请参见[支持的插件列表](#)。

根据提示输入root用户密码。

示例如下：

```
# psql --host=192.168.6.141 --port=5432 --dbname=my_extension_db --  
username=root -c "select control_extension('drop','postgis');"
```

```
Password for user root:  
control_extension  
-----  
drop postgis successfully.  
(1 row)
```

6.11.2 支持的插件列表

📖 说明

如下表格中的数据来源于RDS for PostgreSQL引擎各版本下，最新小版本支持的插件列表。您可以通过**SELECT name FROM pg_available_extensions;**语句，查看当前实例支持的插件列表。

使用mysql_fdw、oracle_fdw、pgsql-ogr-fdw、postgres_fdw和tds_fdw插件时，**要求两个PostgreSQL实例的服务端IP必须在同一个VPC和子网内。**

表 6-11 支持插件列表

插件名称	Postgre SQL 9.5	Postgre SQL 9.6	Postgre SQL 10	Postgre SQL 11	Postgre SQL 增 强版 (1.0版 本)	Postgr eSQL 12
address_standardizer	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
address_standardizer_data_us	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
btree_gin	1.0	1.0	1.2	1.3	1.3	1.3
btree_gist	1.1	1.2	1.5	1.5	1.5	1.5
citext	1.1	1.3	1.4	1.5	1.5	1.6
cube 具体说明请参见 cube 。	1.0	1.2	1.2	1.4	1.4	1.4
dblink	1.1	1.2	1.2	1.2	1.2	1.2
dict_int	1.0	1.0	1.0	1.0	1.0	1.0
dict_xsyn	1.0	1.0	1.0	1.0	1.0	1.0
earthdistance 具体说明请参见 earthdistance 。	1.0	1.1	1.1	1.1	1.1	1.1
fuzzystrmatch	1.0	1.1	1.1	1.1	1.1	1.1
hll	2.12	2.12	2.12	2.12	2.12	2.14
hstore	1.3	1.4	1.4	1.5	1.5	1.6
intagg	1.0	1.1	1.1	1.1	1.1	1.1
intarray	1.0	1.2	1.2	1.2	1.2	1.2
isn	1.0	1.1	1.1	1.2	1.2	1.2
ltree	1.0	1.1	1.1	1.1	1.1	1.1

插件名称	PostgreSQL 9.5	PostgreSQL 9.6	PostgreSQL 10	PostgreSQL 11	PostgreSQL 增强版 (1.0版本)	PostgreSQL 12
oracle_fdw	-	-	1.1	1.1	1.1	1.1
orafce	3.8	3.8	3.8	3.8	3.8	3.8
pg_cron	-	-	-	-	-	1.2
pg_hint_plan	1.1.5	1.2.0	1.3.0	1.3.0	1.3.0	-
pg_jieba	1.1.0	1.1.0	1.1.0	1.1.0	1.1.0	1.1.0
pg_pathman	1.5	1.5	1.5	1.5	1.5	1.5
pg_repack	-	-	-	1.4.6	1.4.6	1.4.6
pgrouting	-	-	-	3.1.0	3.1.0	3.1.0
pg_stat_statements	1.3	1.4	1.6	1.6	1.6	1.7
pg_trgm	1.1	1.3	1.3	1.4	1.4	1.4
pgcrypto	1.2	1.3	1.3	1.3	1.3	1.3
pgstattuple	-	-	1.5	1.5	1.5	1.5
plpgsql	1.0	1.0	1.0	1.0	1.0	1.0
postgis 具体说明请参见 postgis 。	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
postgres_fdw	-	-	1.0	1.0	1.0	1.0
postgis_raster	集成到 postgis	集成到 postgis	集成到 postgis	集成到 postgis	集成到 postgis	3.0.0
postgis_sfcgal	1.3.9	1.3.9	1.3.9	1.3.9	1.3.9	1.3.9
postgis_tiger_geocoder	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
postgis_topology	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
powa-archivist	-	-	-	4.1.2	-	4.1.2
pg_qualstats	-	-	-	2.0.2	-	2.0.2
pg_stat_kcache	-	-	-	2.2.0	-	2.2.0
pg_track_settings	-	-	-	2.0.1	-	2.0.1

插件名称	PostgreSQL 9.5	PostgreSQL 9.6	PostgreSQL 10	PostgreSQL 11	PostgreSQL 增强版 (1.0版本)	PostgreSQL 12
pg_wait_sampling	-	-	-	1.1.3	-	1.1.3
hypopg	-	-	-	1.2.0	-	1.2.0
tablefunc	1.0	1.0	1.0	1.0	1.0	1.0
tds_fdw	-	-	-	2.0.1	2.0.1	2.0.1
timescaledb 具体说明请参见 timescaledb 。	-	1.3.2	1.3.2	1.3.2	1.3.2	1.7.0
unaccent	1.0	1.1	1.1	1.1	1.1	1.1
uuid-osp	1.0	1.1	1.1	1.1	1.1	1.1
wal2json	-	-	-	2.3	2.3	2.3
zhparser	1.0	1.0	1.0	1.0	1.0	1.0

插件说明

- **postgis**
 - 创建postgis插件时，会同步创建以下插件：
postgis
postgis_topology
fuzzystrmatch
postgis_tiger_geocoder
address_standardizer
address_standardizer_data_us
 - 创建postgis插件时，主备实例的主机创建插件后，请先断开备机已经建立的连接，备机需要重新建立连接更新search_path设置。
 - 对于PostgreSQL增强版实例，创建postgis插件前，需要在控制台上将实例参数“empty_is_null”设置为“OFF”。
- **earthdistance**
安装earthdistance插件前，请先安装cube插件。
- **cube**
如果已安装earthdistance插件，删除cube插件会导致earthdistance插件不可用。
- **timescaledb**
RDS for PostgreSQL的timescaledb插件不支持ssl协议的特性，具体请参见[timescaledb插件不支持的部分接口](#)。

- **wal2json**

该插件是逻辑复制插件，您可以直接使用该插件，不需要通过control_extension安装。

timescaledb 插件不支持的部分接口

- add_compress_chunks_policy
- add_drop_chunks_policy
- add_reorder_policy
- alter_job_schedule
- compress_chunk
- decompress_chunk
- drop_chunks
- interpolate
- locf
- move_chunk
- remove_compress_chunks_policy
- remove_drop_chunks_policy
- remove_reorder_policy
- reorder_chunk
- set_integer_now_func
- time_bucket_gapfill

6.12 数据库安全设置

密码复杂度要求

- 关系型数据库实例数据库对在客户端新创建的数据库用户，设置了密码安全策略：
 - 密码最小长度为8个字符。
 - 密码中必须含有字母和非字母，其中，非字母包括数字和特殊字符。
 - 密码不得包含用户名。

创建用户建议

用户在使用CREATE USER或CREATE ROLE命令时，建议指定VALID UNTIL 'timestamp' 参数（timestamp为过期时间戳），设置用户密码的过期时间。

访问数据库对象建议

用户在访问数据库对象时，建议指定数据库对象的schema名，以防止特定场景下的“特洛伊木马”攻击。

账户说明

您在创建PostgreSQL数据库实例时，系统会自动为实例创建如下系统账户（用户不可使用），用于给数据库实例提供完善的后台运维管理服务。

须知

如果试图删掉、重命名、修改这些账户的密码和权限，会导致出错，请谨慎操作。

- rdsAdmin：管理账户，拥有最高的superuser权限，用于查询和修改实例信息、故障排查、迁移、恢复等操作。
- rdsRepl：复制账户，用于备实例或只读实例在主实例上同步数据。
- rdsBackup：备份账户，用于后台的备份。
- rdsMetric：指标监控账户，用于watchdog采集数据库状态数据。
- rdsProxy：数据库代理账户，该账户在开通读写分离时才会自动创建，用于通过读写分离地址连接数据库时鉴权使用。

6.13 数据安全

6.13.1 重置管理员密码

操作场景

关系型数据库服务仅支持通过主实例重置管理员密码。

在使用RDS过程中，如果忘记数据库账号密码，可以重新设置密码。

说明

- 当您修改数据库主实例的密码时，如果该实例中存在备实例或只读实例，则会被同步修改。
- 重置密码生效时间取决于该主实例当前执行的业务数据量。
- 请定期（如三个月或六个月）修改用户密码，以提高系统安全性，防止出现密码被暴力破解等安全风险。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，选择“更多 > 重置密码”。

步骤5 在“重置密码”弹框，输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%^*_+=?;特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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”页签，在“数据库信息”模块的“管理员账户名”处，单击“重置密码”。

步骤6 输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%^*_+=?;特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

6.13.2 修改实例安全组

操作场景

关系型数据库服务支持修改数据库引擎的主实例和只读实例的安全组，对于主备实例，修改主实例的安全组，备实例的安全组会被同步修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择对应的主实例或只读实例，单击实例名称。

步骤5 在“基本信息”页签“连接信息”模块的“安全组”处，单击，选择对应的安全组。

- 单击，提交修改。
- 单击，取消修改。

步骤6 稍后单击“基本信息”页面右上角的，查看修改结果。此过程需1~3分钟。

----结束

6.14 与告警监控指标与告警

6.14.1 支持的监控指标

声明：RDS实例中的Agent只用于监控实例运行的指标、状态，不会收集除监控指标外的其它数据。

功能说明

本节定义了关系型数据库上报云监控的监控指标的命名空间，监控指标列表和维度定义。用户可以通过云监控提供的API接口来检索关系型数据库产生的监控指标和告警信息。

命名空间

SYS.RDS

实例监控指标

- PostgreSQL数据库性能监控指标，如表6-12所示。

表 6-12 数据库性能监控列表

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds001_cpu_util	CPU使用率	该指标用于统计测量对象的CPU使用率，以比率为单位。	0-100%	测量对象： 监控实例类型： PostgreSQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds002_mem_util	内存使用率	该指标用于统计测量对象的内存使用率，以比率为单位。	0-100 %	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds003_iops	IOPS	该指标用于统计当前实例，单位时间内系统处理的I/O请求数量（平均值）。	≥ 0 counts /s	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds004_bytes_in	网络输入吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输入的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds005_bytes_out	网络输出吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输出的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds039_disk_util	磁盘利用率	该指标用于统计测量对象的磁盘利用率，以比率为单位。	0-100 %	测量对象： 监控实例类型： PostgreSQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds040_transaction_logs_usage	事务日志使用量	事务日志所占用的磁盘容量。	≥ 0 MB	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds041_replication_slot_usage	复制插槽使用量	复制插槽文件所占磁盘容量。	≥ 0 MB	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds042_database_connections	数据库连接数	当前连接到数据库的后端量。	≥ 0 counts	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds043_maximum_used_transaction_ids	事务最大已使用ID数	事务最大已使用ID。	≥ 0 counts	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds044_transaction_logs_generations	事务日志生成速率	平均每秒生成的事务日志大小。	≥ 0 MB/s	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds045_oldest_replication_slot_lag	最滞后副本滞后量	多个副本中最滞后副本（依据接收到的WAL数据）滞后量。	≥ 0 MB	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟
rds046_replication_lag	复制时延	副本滞后时延。	≥ 0 ms	测量对象：数据库 监控实例类型：PostgreSQL实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds047_disk_total_size	磁盘总大小	该指标用于统计测量对象的磁盘总大小。	40GB~4000GB	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds048_disk_used_size	磁盘使用量	该指标用于统计测量对象的磁盘使用大小。	0GB~4000GB	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds049_disk_read_throughput	硬盘读吞吐量	该指标用于统计每秒从硬盘读取的字节数。	≥ 0 bytes/s	测量对象： 监控实例类型： PostgreSQL实例	1分钟
rds050_disk_write_throughput	硬盘写吞吐量	该指标用于统计每秒写入硬盘的字节数。	≥ 0 bytes/s	测量对象： 监控实例类型： PostgreSQL实例	1分钟

维度

Key	Value
postgresql_instance_id	PostgreSQL实例ID

6.14.2 设置告警规则

操作场景

通过在云监控中[设置告警规则](#)，用户可自定义关系型数据库的监控目标与通知策略，及时了解关系型数据库运行状况，从而起到预警作用。

设置关系型数据库的告警规则包括设置告警规则名称、服务、维度、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。

设置告警规则

步骤1 登录管理控制台。

步骤2 选择“管理与监管 > 云监控服务 CES”。

步骤3 在左侧导航树栏，选择“告警 > 告警规则”。

步骤4 在“告警规则”界面，单击“创建告警规则”进行添加。

----结束

6.14.3 查看监控指标

操作场景

云服务平台提供的云监控，可以对关系型数据库的运行状态进行日常监控。您可以通过管理控制台，直观地查看关系型数据库的各项监控指标。

由于监控数据的获取与传输会花费一定时间，因此，云监控显示的是当前时间5~10分钟前的关系型数据库状态。如果您的关系型数据库刚创建完成，请等待5~10分钟后查看监控数据。

前提条件

- 关系型数据库正常运行。
故障、删除状态的关系型数据库，无法在云监控中查看其监控指标。当关系型数据库再次启动或恢复后，即可正常查看。

说明

故障24小时及以上的关系型数据库，云监控将默认该关系型数据库不存在，并在监控列表中删除，不再对其进行监控，但告警规则需要用户手动清理。

- 关系型数据库已正常运行一段时间（约10分钟）。
对于新创建的关系型数据库，需要等待一段时间，才能查看上报的监控数据和监控视图。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击操作列中的“查看监控指标”，跳转到云监控服务页面。

您也可以在“实例管理”页面，单击目标实例名称，在页面右上角，单击“查看监控指标”，跳转到云监控服务页面。

步骤5 在云监控页面，可以查看实例监控信息。

云监控支持的性能指标监控时间窗包括：近1小时、近3小时、近12小时、近24小时、近7天。

----结束

6.15 CTS 审计

6.15.1 支持审计的关键操作列表

通过，您可以记录与关系型数据库实例相关的操作事件，便于日后的查询、审计和回溯。

表 6-13 支持的 RDS 操作列表

操作名称	资源类型	事件名称
创建实例、创建只读、恢复到新实例	instance	createInstance
扩容、规格变更	instance	instanceAction
实例重启	instance	instanceRestart
恢复到原有实例	instance	instanceRestore
实例重命名	instance	instanceRename
重置密码	instance	resetPassword
设置数据库版本配置参数	instance	setDBParameters
重置实例的数据库版本配置参数	instance	resetDBParameters
设置备份策略-打开，关闭，修改	instance	setBackupPolicy
修改数据库端口号	instance	changeInstancePort
绑定解绑EIP	instance	setOrResetPublicIP
修改安全组	instance	modifySecurityGroup
创建标签	instance	createTag
删除标签	instance	deleteTag
修改标签	instance	modifyTag
删除实例	instance	deleteInstance
主备倒换	instance	instanceFailOver
修改主备同步模式	instance	instanceFailOverMode
修改主备倒换策略	instance	instanceFailOverStrategy
单机转主备实例	instance	modifySingleToHalfInstance

操作名称	资源类型	事件名称
创建快照	backup	createManualSnapshot
复制快照	backup	copySnapshot
下载备份（通过OBS下载）	backup	downloadSnapshot
下载备份（通过浏览器下载）	backup	backupsDownload
删除快照	backup	deleteManualSnapshot
合并下载binlog	backup	packBackupsDownload
创建参数模板	parameterGroup	createParameterGroup
修改参数模板	parameterGroup	updateParameterGroup
删除参数模板	parameterGroup	deleteParameterGroup
复制参数模板	parameterGroup	copyParameterGroup
重置参数模板	parameterGroup	resetParameterGroup
应用参数模板	parameterGroup	applyParameterGroup
保存参数模板	parameterGroup	saveParameterGroup
冻结删除	all	rdsUnsubscribeInstance
实例冻结	all	rdsfreezeInstance

6.15.2 查看追踪事件

操作场景

在您开通了云审计服务后，系统开始记录云服务资源的操作。云审计服务管理控制台保存最近7天的操作记录。

本节介绍如何在管理控制台查看最近7天的操作记录。

操作步骤

步骤1 登录管理控制台。

步骤2 选择“管理与监管 > 云审计服务 CTS”，进入云审计服务信息页面。

步骤3 单击左侧导航树的“事件列表”，进入事件列表信息页面。

步骤4 事件列表支持通过筛选来查询对应的操作事件。详细信息如下：

- 事件来源、资源类型和筛选类型：在下拉框中选择查询条件。
其中筛选类型选择资源ID时，还需选择或者手动输入某个具体的资源ID。
- 操作用户：在下拉框中选择某一具体的操作用户。
- 事件级别：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- 时间范围：可通过选择时间段查询操作事件。

步骤5 选择查询条件后，单击“查询”。

步骤6 在需要查看的记录左侧，单击展开该记录的详细信息。

步骤7 在需要查看的记录右侧，单击“查看事件”，在弹出框中显示该操作事件结构的详细信息。

步骤8 单击右侧的“导出”，将查询结果以CSV格式的文件导出，该CSV文件包含了云审计服务记录的七天以内的操作事件的所有信息。

关于事件结构的关键字段详解，请参见《云审计服务用户指南》的“事件结构”和“事件样例”章节。

----结束

6.16 任务中心

6.16.1 查看任务

您可以通过“任务中心”查看任务执行进度和结果，并进行管理。

说明

关系型数据库服务支持查看和管理以下任务：

- 创建实例。
- 创建只读实例。
- 单实例转主备。
- 磁盘扩容。
- 实例绑定弹性公网IP。
- 实例解绑弹性公网IP。
- 实例主备倒换。
- 重启实例。
- 恢复到新实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，选择目标任务，查看任务信息。

----结束

6.16.2 删除任务

对于不再需要展示的任务，您可以通过“任务中心”进行任务记录的删除。删除任务仅删除记录，不会删除数据库实例或者停止正在执行中的任务。

须知

删除任务将无法恢复，请谨慎操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，选择目标任务，单击操作列的“删除”，在弹出框中单击“是”，删除任务。

关系型数据库服务支持删除以下状态的即时任务：

- 执行中
- 完成
- 失败

----结束

7 SQL Server 用户指南

7.1 实例生命周期

7.1.1 创建与原实例相同配置的数据库实例

操作场景

关系型数据库服务支持用户快速创建相同配置的数据库实例。

📖 说明

- 创建相同配置实例没有次数限制。
- 只读实例不能执行创建相同配置的操作。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 创建相同配置”。

步骤5 在“购买相同配置”页面，各项配置默认和原实例信息相同，您可根据业务需要进行修改，填选完毕后，单击“立即创建”。

步骤6 确认实例信息。

步骤7 稍后刷新实例列表，查看创建结果。如果实例状态为“正常”，说明实例创建成功。

创建成功后，用户可以在“实例管理”页面对其进行查看和管理。

----结束

7.1.2 重启实例

操作场景

通常出于维护目的，您可能需要重启数据库实例。例如：对于某些运行参数修改，需要重启单个实例使之生效。您可通过控制台对主实例和只读实例执行重启操作。

约束限制

- 如果数据库服务状态异常，可以通过实例的强制重启功能，强制重启服务进程。强制重启会导致数据库服务中未提交的事务强制中断，请谨慎操作。
- 重启数据库实例会重新启动数据库引擎服务。重启数据库实例将导致短暂中断，在此期间，数据库实例状态将显示为“重启中”。
- 重启过程中，实例将不可用。重启后实例会自动释放内存中的缓存，请注意对业务进行预热，避免业务高峰期出现阻塞。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主实例，或者单击  展开只读实例，单击“更多 > 重启实例”。

您也可以在“实例管理”页面单击目标实例名称，进入“基本信息”页面。在页面右上角，单击“重启实例”。

重启实例时，如果是主备实例，对应的备实例也会被同步重启。

步骤5 在“重启实例”弹框，确认实例信息，单击“是”重启实例。

步骤6 稍后刷新实例列表，查看重启结果。如果实例状态为“正常”，说明实例重启成功。

----结束

7.1.3 自定义列表项

操作场景

您可以根据自身业务需要，自定义设置实例信息列表项。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击 ，选择自定义列表项。

- 系统默认展示以下列表项：实例名称/ID、实例类型、数据库引擎版本、运行状态、内网地址、操作。
默认实例信息列表项不可更改。
- 单project情况下，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。
- 多project情况下，需要同时开通项目管理服务权限，您最多可选择9个列表项：实例名称/ID、实例类型、数据库引擎、运行状态、内网地址、创建时间、数据库端口、存储空间类型、操作。

----结束

7.1.4 导出实例

操作场景

您可以导出所有实例，或根据一定条件筛选出来的目标实例，查看并分析实例信息。

导出所有实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例列表右上角 ，默认导出所有的数据库实例，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

导出筛选的目标实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，根据引擎类型、实例名称、实例ID、实例内网地址等条件，筛选实例，或勾选需要导出的实例，单击实例列表右上角 ，在导出弹框勾选所需导出信息，单击“导出”。

步骤5 导出任务执行完成后，您可在本地查看到一个“.csv”文件。

----结束

7.1.5 删除按需实例

操作场景

您可根据业务需要，在关系型数据库“实例管理”页面手动删除实例来释放资源。

约束限制

- 正在执行操作的实例不能手动删除，只有在实例操作完成后，才可删除实例。
- 实例删除后将不再产生费用，实例生成的自动备份会被同步删除，保留的手动备份会继续收取费用。

须知

- 删除主实例时，会同步删除其对应的备实例和只读实例，请您谨慎操作。
- 实例删除后，资源立即释放，请谨慎操作。如需保留数据，请务必确认完成[数据备份](#)后再删除实例。
- 通过保留的手动备份，可以恢复实例数据，具体请参见[通过备份文件恢复关系型数据库实例数据](#)。

删除主实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择需要删除的主实例，在“操作”列，选择“更多 > 删除实例”。

步骤5 在“删除实例”弹框，单击“是”下发请求。

步骤6 稍后刷新“实例管理”页面，查看删除结果。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面的实例列表中，选择对应主实例，单击实例名称前的 ，可查看到该主实例下的全部只读实例。

步骤5 选择目标只读实例，在“操作”列，选择“更多 > 删除实例”。

步骤6 在“删除实例”弹框，单击“是”下发请求。

步骤7 稍后刷新“实例管理”页面，查看删除结果。

----结束

7.1.6 回收站

操作场景

RDS支持将删除的主备或者单机实例，加入回收站管理。通过在回收站中重建实例，将数据恢复到新实例上。新实例数据库引擎、数据库版本、存储类型与原实例相同，其他参数可以重新配置。默认可以恢复1~7天内删除的实例。

约束限制

- RDS不回收只读实例，只有主备或者单机实例才会进入回收站。
- 回收站策略机制默认开启，且不可关闭。

设置回收站策略

须知

修改回收站保留天数，仅对修改后新进入回收站的实例生效，对于修改前已经存在的实例，仍保持原来的回收策略，请您谨慎操作。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在左侧导航栏，单击“回收站”。

步骤5 在“回收站”页面，单击“回收站策略”，设置已删除实例保留天数，可设置范围为1~7天。

步骤6 单击“确定”，完成设置。

----结束

重建实例

在回收站保留期限内的主实例可以通过重建实例恢复数据。

步骤1 在左侧导航栏，单击“回收站”。

步骤2 在“回收站”页面，在实例列表中找到需要恢复的目标实例，单击操作列的“重建”。

步骤3 在“重建新实例”页面，选填配置后，提交重建任务，具体可参考[恢复到新实例操作](#)。

----结束

7.2 变更实例

7.2.1 修改实例名称

操作场景

关系型数据库服务支持修改主实例和只读实例的实例名称，以方便用户识别。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击目标实例名称后的 ，编辑实例名称，单击“确认”，即可修改实例名称。

您也可以单击目标实例名称，进入实例的“基本信息”页面，在“数据库信息”模块实例名称处，单击 ，修改实例名称。

实例名称长度最大64个字符，必须以字母开头，可包含大写字母、小写字母、数字、中划线、下划线，不能包含其他特殊字符。

- 单击 ，提交修改。
- 单击 ，取消修改。

步骤5 在实例的“基本信息”页面，查看修改结果。

----结束

7.2.2 切换主备实例的可用性策略

操作场景

关系型数据库服务支持切换主备实例的可用性策略，以满足不同业务需求。可选择“可靠性优先”或者“可用性优先”两种策略。

- 可靠性优先（默认选择）：数据库应该尽可能保障数据的可靠性，即数据丢失量最少。对于数据一致性要求较高的业务，建议选择该策略。
- 可用性优先：数据库应该尽可能快速恢复服务，即可用时间最长。对于数据库在线时间要求较高的业务，建议选择该策略。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击主实例名称。

步骤5 在“基本信息”页面，在“数据库信息”模块的“切换策略”处，单击“修改”，在弹出框中，选择策略，单击“确定”，完成修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

7.2.3 变更实例的 CPU 和内存规格

操作场景

CPU/内存规格可根据业务需要进行变更，当实例的状态由“规格变更中”变为“正常”，则说明变更成功。

约束限制

- 当实例进行CPU/内存规格变更时，该实例不可被删除。
- 修改CPU/内存后，将会重启数据库实例。请选择业务低峰期，避免业务异常中断。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击“操作”列的“更多 > 规格变更”，进入“规格变更”页面。

您也可以通过单击目标实例名称，进入“基本信息”页面，在“数据库信息”模块的“性能规格”处，单击“规格变更”，进入“规格变更”页面。

步骤5 在“规格变更”页面，选择所需修改的性能规格，单击“下一步”。

步骤6 查看变更结果。

任务提交成功后，单击“返回实例列表”，在实例管理页面，可以看到实例状态为“规格变更中”。稍后在对应的“基本信息”页面，查看实例规格，检查修改是否成功。此过程需要5~15分钟。

须知

RDS SQL Server规格变更后，数据库参数“max server memory”自动调整。max server memory值建议设置为内存大小-520MB，例如：您的内存为1GB，即1024MB，那么，该值设置为1GB-520MB，即504MB。

----结束

7.2.4 磁盘扩容

操作场景

随着业务数据的增加，原来申请的数据库磁盘容量可能会不足，需要为关系型数据库实例进行扩容。

当实例处于“磁盘空间满”状态时，需扩容至磁盘空间使用率小于85%才可使实例处于可用状态，使数据库恢复正常的写入操作。建议您设置“磁盘使用率”指标的告警规则，及时了解磁盘使用情况，起到预警作用，具体请参见[设置告警规则](#)。

数据库磁盘空间不足的原因和解决方案，请参见[数据超过了RDS实例的最大存储容量怎么办](#)。

磁盘扩容期间，服务不中断。关系型数据库服务只支持扩容磁盘存储空间大小，不支持变更存储类型。

约束限制

- 关系型数据库实例支持无限次扩容。
- 云数据库SQL Server实例最大可扩容至4000GB，扩容次数没有限制。
- 如果是主备实例，针对主实例扩容时，会同时对其备实例进行扩容。
- 扩容过程中，该实例不可重启和删除。
- 磁盘容量变更只允许扩容，不能缩容。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 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。

您还可以通过如下途径进行扩容：

- 单击目标实例名称，进入“基本信息”页面。在“存储空间”模块，单击“磁盘扩容”，进入“磁盘扩容”页面。

步骤5 在“服务选型”页面，选择空间大小，“下一步”。

每次扩容最小容量为10GB，实例所选容量大小必须为10的整数倍，最大磁盘容量为4000GB。

步骤6 规格确认。

- 重新选择：单击“上一步”，回到上个页面，修改新增大小。
- 确认无误：单击“提交申请”，提交扩容。
- 确认无误：单击“提交”，提交扩容。

步骤7 查看扩容结果。

在实例管理页面，可看到实例状态为“扩容中”，稍后单击实例名称，在“基本信息”页面，查看磁盘大小，检查扩容是否成功。此过程需要3~5分钟。

----结束

7.2.5 单机转主备实例

操作场景

- 关系型数据库支持数据库单机实例转为主备实例。在保留原实例资源的情况下提高了实例的可靠性。
- 主备实例可实现自动故障倒换，备机快速接管业务。建议您将主备实例选择到不同的可用区，享受跨可用区，同城容灾的高可用服务。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择单机实例，单击“操作”列的“更多 > 转主备”，进入“转主备”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例左侧的添加按钮



，进入“转主备”页面。

步骤5 在转主备时，您需选择“备可用区”，输入“原实例管理员密码”，其他信息默认与主实例相同。确认信息无误，单击“提交”。

步骤6 单机转主备创建成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 创建过程中，状态显示为“转主备中”。您可以通过“任务中心”查看详细进度。具体请参见[任务中心](#)。

- 在实例列表的右上角，单击  刷新列表，可查看到单机转主备完成后，实例状态显示为“正常”。“实例类型”显示为“主备”。

----结束

7.2.6 手动切换主备实例

操作场景

当关系型数据库主备类型的实例创建成功后，系统会在同一区域内为该实例创建一个备实例。主备实例的数据会实时同步，用户仅能访问主实例数据，备实例仅作为备份形式存在。您可根据业务需要，进行主备实例的切换，切换后原来的主实例会变成备实例，可实现机架级别的容灾。

约束限制

- 实例运行正常。
- 主备实例复制关系正常。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在“基本信息”页面中“数据库信息”模块的“实例类型”处，单击“主备切换”。

您也可以在“基本信息”页面，单击“实例拓扑图”模块的 。进行主备切换。

须知

主备切换可能会造成几秒或几分钟的服务闪断（闪断时间与复制时延有关），并有可能在主备同步时延过大的情况下，导致少量数据丢失。请在业务低峰期进行主备切换，避免对高峰期业务造成影响。

步骤6 在“主备切换”弹框，单击“是”进行主备实例的切换。

在“复制状态”为“正常”的情况下，如果复制时延大于300s，主备切换任务无法下发。

步骤7 主备切换成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行检查和管理。

- 切换过程中，状态显示为“主备切换中”。
- 在实例列表的右上角，单击  刷新列表，可查看到主备切换完成后，实例状态显示为“正常”。

----结束

7.3 只读实例

7.3.1 只读实例简介

产品简介

目前，云数据库SQL Server仅2019企业版和2017企业版的实例支持新增只读实例。

在对数据库有少量写请求，但有大量读请求的应用场景下，单个实例可能无法抵抗读取压力，甚至对主业务产生影响。为了实现读取能力的弹性扩展，分担数据库压力，您可以在某个区域中创建一个或多个只读实例，利用只读实例满足大量的数据库读取

需求，以此增加应用的吞吐量。您需要在应用程序中分别配置主实例和每个只读实例的连接地址，才能实现将写请求发往主实例而将读请求发往只读实例。

只读实例为单个节点的架构（没有备节点），采用SQL Server的原生复制功能，将主实例的更改同步到所有只读实例，而且主实例和只读实例之间的数据同步不受网络延时的影响，只读实例跟主实例在同一区域，但可以在不同的可用区。

功能特点

- 规格可以与主实例不一致，并可以随时更改规格（没有时间限制），便于弹性升降级。
- 不需要维护账号与数据库，全部通过主实例同步。
- 提供系统性能监控。
关系型数据库服务提供近20个系统性能的监控视图，如磁盘容量、IOPS、连接数、CPU利用率、网络流量等，用户可以轻松查看实例的负载。
- 备份设置：不支持自动备份设置以及手动备份。
- 实例恢复：不支持通过备份文件或任意时间点创建只读实例，不支持通过备份集覆盖只读实例。
- 数据迁移：不支持将数据迁移至只读实例。

功能限制

1个主实例最多可以创建5个只读实例。

创建和管理只读实例

- [创建只读实例](#)
- [管理只读实例](#)

7.3.2 创建只读实例

操作场景

只读实例用于增强主实例的读能力，减轻主实例负载。

关系型数据库实例创建成功后，可根据业务需要创建只读实例。

说明

一个主实例中，最多可以增加5个只读实例。

目前，云数据库SQL Server仅2019企业版和2017企业版的实例支持新增只读实例。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击“更多 > 创建只读”，进入“服务选型”页面。

您也可在实例的“基本信息”页面，单击实例拓扑图中，主实例下方的添加按钮，创建只读实例。

步骤5 在“服务选型”页面，填选实例相关信息后，单击“立即创建”。

表 7-1 基本信息

参数	描述
当前区域	只读实例默认与主实例在同一区域。
实例名称	实例名称的长度在4~64个字符之间，必须以字母开头，可包含大写字母、小写字母、数字、中划线或下划线，不能包含其他特殊字符。
数据库引擎	默认与主实例的数据库引擎一致，不可更改。
数据库版本	默认与主实例的数据库版本一致，不可更改。
可用区	关系型数据库服务支持在同一个可用区内，或者跨可用区部署数据库主实例和只读实例，以提高可靠性。

表 7-2 规格与存储

参数	描述
性能规格	实例的CPU和内存。不同性能规格对应不同连接数和最大IOPS。 关于性能规格详情，请参见 数据库实例规格 。 创建成功后可进行规格变更，请参见 变更实例的CPU和内存规格 。
存储类型	实例的存储类型决定实例的读写速度。最大吞吐量越高，读写速度越快。 <ul style="list-style-type: none">高I/O：最大吞吐量150MB/s超高I/O：最大吞吐量350MB/s
存储空间	您申请的存储空间会有必要的文件系统开销，这些开销包括索引节点和保留块，以及数据库运行必需的空间。 只读实例的存储空间大小默认与主实例一致。

表 7-3 网络

参数	描述
虚拟私有云	和主实例相同。
子网	和主实例相同，创建只读实例时RDS会自动为您配置内网地址，您也可输入子网号段内未使用的内网地址，实例创建成功后该内网地址暂不可修改。
安全组	和主实例相同。

步骤6 在“规格确认”页面，进行信息确认。

- 如果需要重新选择，单击“上一步”，回到服务选型页面修改基本信息。
- 信息确认无误，单击“提交”，下发新增只读实例请求。

步骤7 只读实例创建成功后，用户可以在“实例管理”页面，选择只读实例所对应的实例，单击对其进行查看和管理。

您也可以在基本信息页面的“实例拓扑图”中，单击只读实例的名称，进入该只读实例的“基本信息”页面，对其进行查看和管理。

----结束

后续操作

管理只读实例

7.3.3 管理只读实例

通过只读实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的，单击目标只读实例的名称，进入实例的基本信息页面，即进入只读实例的管理页面。

----结束

通过主实例管理

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 单击只读实例所在的主实例名称，进入主实例的“基本信息”页面。

步骤5 在“实例拓扑图”中单击只读实例的名称，进入该只读实例的“基本信息”页面，即进入只读实例的管理页面。

----结束

删除只读实例

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在实例列表中，单击实例名称前的∨，选择目标只读实例，单击操作列中的“更多 > 删除实例”。

----结束

7.4 备份与恢复

7.4.1 备份概述

关系型数据库支持数据库实例的备份和恢复，以保证数据可靠性。

自动备份

关系型数据库服务会在数据库实例的备份时段中创建数据库实例的自动备份。系统根据您的指定备份保留期保存数据库实例的自动备份。如果需要，您可以将数据恢复到备份保留期中的任意时间点。具体请参见[设置自动备份策略](#)。

手动备份

您还可以创建手动备份对数据库进行备份，手动备份是由用户启动的数据库实例的全量备份，手动备份会一直保存，直到用户手动删除。具体请参见[创建手动备份](#)。

下载备份文件

您可以下载全量备份文件，用于本地存储备份或者恢复数据库。具体请参见[下载备份文件](#)。

7.4.2 设置自动备份策略

操作场景

创建关系型数据库实例时，系统默认开启自动备份策略，安全考虑，实例创建成功后不可关闭，您可根据业务需要设置自动备份策略，关系型数据库服务按照您设置的自动备份策略对数据库进行备份。

关系型数据库服务的备份操作是实例级的，而不是数据库级的。当数据库故障或数据损坏时，可以通过备份恢复数据库，从而保证数据可靠性。备份以压缩包的形式存储在对象存储服务上，以保证用户数据的机密性和持久性。由于开启备份会损耗数据库读写性能，建议您选择业务低峰时间段设置自动备份。

关系型数据库默认开启的自动备份策略设置如下：

- 保留天数：默认为7天。
- 备份时间段：默认为24小时中，间隔一小时的随机的一个时间段，例如01:00 ~ 02:00，12:00 ~ 13:00等。备份时间段以UTC时区保存。如果碰到夏令时/冬令时切换，备份时间段会因时区变化而改变。
- 备份周期：默认为一周内的每一天。

查看或修改自动备份策略

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏，单击“备份恢复”，单击“修改备份策略”。您可以查看到已设置的备份策略，如需修改备份策略，请调整以下参数的值。

- 保留天数是指自动备份可保留的时间，增加保留天数可提升数据可靠性，请根据需要设置。
- 减少保留天数的情况下，该备份策略对已有备份文件同时生效，即超出备份保留天数的已有备份文件会被删除。
- 保留天数为全量自动备份和增量备份的保留时长，范围为1~732天，备份时间段为间隔1小时，建议根据业务情况，选择业务低峰时段，备份周期默认全选，可修改，且至少选择一周中的一天。

步骤6 单击“确定”，确认修改。

----结束

7.4.3 创建手动备份

操作场景

关系型数据库服务支持对运行正常的主实例创建手动备份，用户可以通过手动备份恢复数据，从而保证数据可靠性。

说明

当数据库实例被删除时，关系型数据库实例的自动备份将被同步删除，手动备份不会被删除。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，在操作列选择“更多 > 创建备份”。

步骤5 在创建备份弹出框中，命名该备份，在备份数据库下拉列表中选择用户创建的数据库，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!<'&=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

📖 说明

系统数据库会默认进行备份。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏，单击“备份恢复”，对其进行查看并管理。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“备份恢复”，单击“创建备份”，命名该备份，在备份数据库下拉列表中选择用户创建的数据库，并添加描述，单击“确定”，提交备份创建，单击“取消”，取消创建。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，且不能包含回车和>!"&'=特殊字符。
- 手动备份创建所需时间由数据量大小决定。

📖 说明

系统数据库会默认进行备份。

步骤6 手动备份创建成功后，用户可在“备份管理”页面，对其进行查看并管理。

也可在“实例管理”页面，单击实例名称，在左侧导航栏中选择“备份恢复”，对其进行查看并管理。

----结束

7.4.4 下载备份文件

操作场景

用户可以下载手动备份和自动备份文件，用于本地存储备份或者恢复数据库。

按地址下载

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要下载的可用备份，单击操作列中的“下载”。

您也可进入目标实例的“基本信息”页面，在左侧导航栏选择“备份恢复”，单击操作列中的“下载”。

步骤5 在弹出框中选择下载方式。

在弹出框中，单击“按地址下载”，单击复制地址，通过下载地址下载备份文件。

对于SQL Server数据库实例，显示当前备份对应所有备份文件的下载地址，即支持用户下载单个数据库的备份文件。

- 您可以使用其他下载工具下载备份文件。
- 您还可以使用wget命令下载备份文件：

```
wget -O FILE_NAME --no-check-certificate "DOWNLOAD_URL"
```

命令中的参数解释如下：

FILE_NAME：下载成功后的备份文件名称，由于原始文件名称较长，可能会超出客户端文件系统的限制，建议下载备份文件时使用“-O”进行重命名。

DOWNLOAD_URL：需下载的备份文件所在路径，如果包含特殊字符则需要转义。

---结束

7.4.5 通过备份文件恢复关系型数据库实例数据

操作场景

关系型数据库支持使用已有的自动备份和手动备份，将实例数据恢复到备份被创建时的状态。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“备份管理”页面，选择需要恢复的备份，单击操作列的“恢复”。

您也可在“实例管理”页面，单击指定的实例名称，在左侧导航栏单击“备份恢复”，在“全量备份”页签下单击目标备份对应的操作列中的“恢复”。

步骤5 在“恢复到指定时间点”弹出框中，填选相关信息，单击“确定”。

1. 选择需要的恢复方式。

- 新实例：将备份恢复到新创建的实例上。

跳转到“恢复到新实例”的服务选型页面：

- 数据库引擎与原实例相同，不可修改。
- 存储空间大小默认和原实例相同，且必须大于或等于原实例存储空间大小。

- 当前实例：将备份数据恢复到当前实例上。

须知

- 如果备份所在的数据库实例已被删除，则不可恢复到当前实例。
 - 恢复到当前实例会导致当前实例的全部数据被覆盖，并且恢复过程中数据库不可用。
-
- 恢复到已有实例：将备份数据恢复到已经存在的实例上。

须知

- 恢复到已有实例会导致实例数据被覆盖，且恢复过程中实例将不可用。
- 只可选择与原实例相同VPC，相同引擎，相同版本或高版本的实例。
- 请确保目标实例的存储空间大于或等于当前实例的存储空间，否则会导致任务下发失败。
- 如果原实例开启TDE，不允许恢复到已有实例。

选择目标实例，单击“下一步”。

2. 勾选需要恢复的数据库，您可以根据需要，自定义恢复后的新数据库名。如果恢复到的新数据库名未填写，默认按原数据库名进行恢复。

说明

- 新数据库名不能与原数据库重名，重命名恢复之后，使用原库名建立的同义词不可用。
- 新数据库名不能包含rdsadmin、master、msdb、tempdb、model或resource字段（不区分大小写）。
- 数据库名称长度在1~64个字符之间，包含字母、数字、下划线或中划线，不能包含其他特殊字符。

步骤6 查看恢复结果。**● 恢复到新实例**

RDS会为您重新创建一个和该备份数据相同的实例。可看到实例由“创建中”变为“正常”，说明恢复成功。

恢复成功的新实例是一个独立的实例，与原有实例没有关联。

● 恢复到当前实例

在“实例管理”页面，可查看目标实例状态为“恢复中”，恢复完成后，实例状态由“恢复中”变为“正常”。如果目标实例下存在只读实例，只读实例的状态与目标实例一致。

恢复成功后，会执行一次全量备份。

● 恢复到已有实例

在“实例管理”页面，可查看该实例下所有实例状态为“恢复中”，恢复完成后，实例状态由“恢复中”变为“正常”。

----结束

7.4.6 复制备份

操作场景

关系型数据库服务支持复制自动和手动备份，制作一个不同名的拷贝。

限制条件

只支持在同一区域下复制并使用备份，暂不支持跨region操作。

备份保留策略

- 如果需要长期保留自动备份，您可以复制该自动备份生成一个手动备份，该手动备份在您删除之前将会一直保留。
- 如果手动备份所使用的空间超过您的默认存储空间，则可能会增加关系型数据库存储成本。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入“基本信息”页面，在左侧导航栏，单击“备份恢复”，选择目标备份，单击操作列中的“复制”。

步骤5 填写新备份名称和描述，单击“确定”。

- 备份名称的长度在4~64个字符之间，必须以字母开头，区分大小写，可以包含字母、数字、中划线或者下划线，不能包含其他特殊字符。
- 备份描述不能超过256个字符，不能包含回车和>!<"&'=特殊字符。

步骤6 新备份创建成功后，用户可在“备份管理”页面，查看并管理自己的备份。

----结束

7.5 参数模板管理

7.5.1 创建参数模板

您可以使用数据库参数模板中的参数来管理数据库引擎配置。数据库参数模板就像是引擎配置值的容器，这些值可应用于一个或多个数据库实例。

默认参数模板包含数据库引擎默认值和系统默认值，具体根据引擎、计算等级及实例的分配存储空间而定。您无法修改默认数据库参数模板的参数设置，您必须创建自己的数据库参数模板才能更改参数设置的默认值。

须知

并非所有数据库引擎参数都可在客户创建的数据库参数模板中进行更改。

如果您想使用您自己的数据库参数模板，只需创建一个新的数据库参数模板，创建实例的时候选择该参数模板，如果是在创建实例后有这个需求，可以重新应用该参数模板，请参见[应用参数模板](#)。

若您已成功创建数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案，请参见[复制参数模板](#)。

以下是您在使用数据库参数模板中的参数时应了解的几个要点：

- 当您更改动态参数并保存数据库参数模板时，将立即应用更改。当您更改静态参数并保存数据库参数模板时，参数更改将在您手动重启数据库实例后生效。
- 在数据库参数模板内设置参数不恰当可能会产生意外的不利影响，包括性能降低和系统不稳定。修改数据库参数时应始终保持谨慎，且修改数据库参数模板前要备份数据。将参数模板更改应用于生产数据库实例前，您应当在测试数据库实例上试用这些参数模板设置更改。

说明

关系型数据库和文档数据库服务不共享参数模板配额。

每个用户最多可以创建100个关系型数据库参数模板，各关系型数据库引擎共享该配额。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，单击“创建参数模板”。

步骤5 选择数据库引擎版本，命名参数模板并添加对该参数模板的描述，单击“确定”，创建参数模板。

- 选择该数据库引擎参数模板所需应用的参数模板类型。
- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<"&'=特殊字符。

----结束

7.5.2 编辑参数模板

为确保关系型数据库服务发挥出最优性能，用户可根据业务需求对用户创建的参数模板中的参数进行调整。

您可以修改用户创建的数据库参数模板中的参数值，但不能更改默认数据库参数模板中的参数值。

如果您更改一个参数值，则所做更改的应用时间将由该参数的类型决定。

关系型数据库服务的管理控制台显示与数据库实例关联的数据库参数模板的状态。例如，如果数据库实例未使用对其关联的数据库参数模板所做的最新更改，则关系型数据库服务的管理控制台将显示状态为“等待重启”的数据库参数模板。您将需要手动重启数据库实例，以使最新的参数更改对该数据库实例生效。

📖 说明

系统提供的默认参数模板不允许修改，只可单击参数模板名进行查看。当用户参数设置不合理导致数据库无法启动时，可参考默认参数模板重新配置。

批量修改参数

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，选择需要编辑的自定义参数模板，单击参数模板名称。

步骤5 默认在“参数”页签下，根据需要修改相关参数。

相关参数说明如下：

- 参数“remote access”，将此选项设置为0（默认值），表示阻止本地存储过程在远程服务器上执行，或远程存储过程在本地服务器上执行。
- 参数“max server memory”（MB），服务器内存选项，默认值(MB)=操作系统内存(MB)-520(MB)。可为它指定的最小值为1024MB。

可进行的操作如下：

- 单击“保存”，在弹出框中单击“确定”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

步骤6 参数修改完成后，您可以单击“参数修改历史”查看参数的修改详情。

步骤7 参数模板修改后，不会立即应用到当前使用的实例，您需要进行应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤8 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

---结束

修改当前实例的参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

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 在左侧导航栏中选择“参数修改”，在“参数”页签修改相应参数。

参数相关说明如下：

- 参数“remote access”，将此选项设置为0（默认值），表示阻止本地存储过程在远程服务器上执行，或远程存储过程在本地服务器上执行。
- 参数“max server memory”（MB），服务器内存选项，默认值(MB)=操作系统内存(MB)-520(MB)。可为它指定的最小值为1024MB。

可进行的操作如下：

须知

根据参数列表中“是否需要重启”提示，进行相应操作：

- 是：在实例列表中，查看“运行状态”，如果显示“参数变更，等待重启”，则需重启实例使之生效。
 - 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
 - 修改只读实例的某些参数，需要重启该只读实例使之生效。
- 否：无需重启，立即生效。

- 单击“保存”，在弹出框中单击“确定”，保存修改。
- 单击“取消”，放弃本次设置。
- 单击“预览”，可对比参数修改前和修改后的值。

参数修改完成后，您可在[查看参数修改历史](#)。

---结束

7.5.3 导出参数模板

操作场景

- 您可以将您创建的数据库实例参数列表，导出并生成一个新的参数模板，供您后期使用。请参考[应用参数模板](#)将导出的参数模板应用到新的实例。
- 您可以将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV中，方便查看并分析。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”页签单击“导出”。

- 导出到参数模板。将该实例对应参数列表导出并生成一个参数模板，供您后期使用。

在弹出框中，填写新参数模板名称和描述，单击“确定”。

📖 说明

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!"&'=特殊字符。

创建完成后，会生成一个新的参数模板，您可在“参数模板管理”页面的对其进行管理。

- 导出到文件。将该实例对应的参数模板信息（参数名称，值，描述）导出到CSV表中，方便用户查看并分析。

在弹出框中，填写文件名称，单击“确定”。

📖 说明

文件名称长度在4~64个字符之间，必须以字母开头，可以包含字母、数字、中划线或下划线，不能包含其他特殊字符。

----结束

7.5.4 比较参数模板

操作场景

您可以比较同数据库类型的实例参数模板，以了解当前实例参数的差异项。

您也可以比较同数据库类型的默认参数模板，以了解当前参数模板的配置情况。

比较当前实例参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，单击实例名称，进入实例的“基本信息”页签。

步骤5 在左侧导航栏中选择“参数修改”，在“参数”子页签中单击“比较参数”。

步骤6 在弹出框中选择当前实例同数据库类型的参数模板，单击“确定”，比较两个参数的差异项。

- 有差异项，则会显示差异参数的如下信息：参数名称、当前实例参数模板的参数值和被比较参数模板的参数值。
- 无差异项，则不显示。

----结束

比较目标参数模板

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“比较”。

步骤5 选择同一数据库引擎的不同参数模板，单击“确定”，比较两个参数模板之间的配置参数差异项。

- 有差异项，则会显示差异参数模板的如下信息：参数名称、两个参数模板的参数值。
- 无差异项，则不显示。

----结束

7.5.5 查看参数修改历史

操作场景

您可以查看当前实例所使用参数模板以及自定义参数模板的修改历史，以满足业务需要。

说明

用户创建或导出的新参数模板，在未进行参数修改前，无修改历史。

查看当前实例的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例的基本信息页面。

步骤5 在左侧导航栏，单击“参数修改”，单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态、修改时间、是否应用以及应用时间。

如修改后参数模板未应用，请根据业务需要，参考[应用参数模板](#)，将其应用到对应实例。

----结束

查看目标参数模板的参数修改历史

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，单击目标参数模板名称。

步骤5 单击“参数修改历史”。

您可查看参数对应的参数名称、修改前参数值、修改后参数值、修改状态和修改时间。

----结束

7.5.6 复制参数模板

操作场景

您可以复制您创建的自定义数据库参数模板。当您已创建一个数据库参数模板，并且想在新的数据库参数模板中包含该组中的大部分自定义参数和值时，复制参数模板是一个方便的解决方案。

复制数据库参数模板之后，新参数模板可能不会立即显示，建议您等待5分钟再使用。

您无法复制默认参数模板。不过，您可以创建基于默认参数模板的新参数模板。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要复制的参数模板，单击“复制”。

您还可以在“实例管理”页面，单击实例名称，在左侧导航栏，单击“参数修改”，单击“导出”，将该实例对应参数列表导出并生成一个参数模板，供您后期使用。

步骤5 在弹出框中，填写新参数模板名称和描述，单击“确定”。

- 参数模板名称长度在1~64个字符之间，区分大小写，可包含字母、数字、中划线、下划线或句点，不能包含其他特殊字符。
- 参数模板的描述长度不能超过256个字符，且不能包含回车和>!<''&'=特殊字符。

创建完成后，会生成一个新的参数模板，您可在参数模板列表中对其进行管理。

----结束

7.5.7 重置参数模板

操作场景

您可根据自己的业务需求，重置自己创建的参数模板对应的所有参数，使其恢复到默认值。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要设置的参数模板，单击“更多 > 重置”。

步骤5 单击“是”，重置所有参数为其默认值。

步骤6 参数模板修改后，不会立即应用到当前使用的实例，您需要进行应用操作才可生效，具体操作请参见[应用参数模板](#)。

步骤7 应用参数模板后，在实例列表中，查看“运行状态”。

如果显示“参数变更，等待重启”，则需重启实例使之生效；否则，无需重启。

- 由于变更规格导致的强制重启，不会触发该参数生效。
- 修改主实例的某些参数（如果是主备实例，备实例的参数也会被同步修改），需重启主实例使之生效。
- 修改只读实例的某些参数，需要重启该只读实例使之生效。

----结束

7.5.8 应用参数模板

操作场景

参数模板编辑修改后，您可以根据业务需要应用到实例中，参数模板只能应用于相同版本的实例中。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面，根据参数模板类型不同进行如下操作。

- 若需要将默认参数模板应用到实例，在“系统默认”页签的目标参数模板单击“应用”。
- 若需要将用户自己创建的参数模板应用到实例，在“自定义”页签的目标参数模板单击“更多 > 应用”。

一个参数模板可被应用到一个或多个实例。

步骤5 在弹出框中，选择或输入所需应用的实例，单击“确定”。

参数模板应用成功后，您可[查看参数模板应用记录](#)。

----结束

7.5.9 查看参数模板应用记录

操作场景

参数模板编辑修改后，您可根据业务需要将其应用到对应实例中，RDS支持查看参数模板所应用到实例的记录。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 单击“参数模板管理”。

步骤5 在“系统默认”页签下，选择目标参数模板，单击“应用记录”；或在“自定义”页签下，选择目标参数模板，单击“更多 > 应用记录”，查看应用记录。

您可查看参数模板所应用到的实例名称/ID、应用状态、应用时间、失败原因。

----结束

7.5.10 修改参数模板描述

操作场景

参数模板创建成功后，用户可根据需要对自己创建的参数模板描述进行修改。

说明

默认参数模板的描述不可修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择一个用户创建的参数模板，单击“描述”列 。

步骤5 输入新的描述信息，单击 ，提交修改，单击，取消修改。

- 参数模板的描述长度不能超过256个字符，且不能包含>!<"&'=特殊字符。
- 修改成功后，可在参数模板列表的“描述”列查看改后的描述信息。

----结束

7.5.11 删除参数模板

操作场景

您可删除废弃的参数模板。

须知

- 参数模板删除后，不可恢复，请谨慎操作。
- 默认参数模板不可被删除。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“参数模板管理”页面的“自定义”页签，选择需要删除的参数模板，单击“更多 > 删除”。

步骤5 单击“是”，删除参数模板。

----结束

7.6 连接管理

7.6.1 配置和修改内网地址

操作场景

用户从线下或者其他云迁移到关系型数据库后要面对更改IP的问题，为减少客户业务更改，降低迁移难度。提供规划与更改内网IP方式，降低客户迁移成本。

约束限制

修改内网IP后域名需要几分钟重新解析地址导致数据库连接中断，请在业务停止期间操作。

配置内网 IP

在购买实例时，可在“服务选型”页面的网络部分，根据选择的子网自动配置内网地址。

修改内网 IP

对于创建完成的关系型数据库实例，支持更改内网地址。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 在“基本信息”页“连接信息”模块“内网地址”处，单击“修改”。

步骤6 在“修改内网地址”弹出框中进行修改。单击“确定”，保存修改内容。

已使用IP地址列表中使用情况为“待使用”的IP地址表示已被占用，不允许被当前实例使用。

----结束

7.6.2 绑定和解绑弹性公网 IP

操作场景

关系型数据库实例创建成功后，默认未开启公网访问功能（即未绑定弹性公网IP）。关系型数据库服务支持用户绑定弹性公网IP，在公共网络来访问数据库实例，绑定后也可根据需要解绑。

须知

为保证数据库可正常访问，请确保数据库使用的安全组开通了相关端口的访问权限，假设数据库的访问端口是8635，那么需确保安全组开通了8635端口的访问。

前提条件

- 只有主实例和只读实例才能绑定弹性公网IP。
- 对于已绑定弹性公网IP的实例，需解绑后，才可重新绑定其他弹性公网IP。

绑定弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，单击实例名称，进入实例基本信息页面。

步骤5 选择“弹性公网IP”页面，单击“绑定弹性公网IP”。

步骤6 在弹出框的EIP地址列表中，显示“未绑定”状态的EIP，选择所需绑定的EIP，单击“确定”，提交绑定任务。如果没有可用的EIP，单击“查看弹性公网IP”，获取EIP。

步骤7 在“弹性公网IP”页面，查看绑定成功的EIP。

您也可以在“任务中心”页面，查看绑定弹性公网IP任务的执行进度及结果。

如需关闭，请参见[解绑弹性公网IP](#)。

----结束

解绑弹性公网 IP

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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 对于已绑定EIP的实例，在“实例管理”页面，选择指定实例，单击实例名称，进入实例基本信息页面。
- 步骤5** 选择“弹性公网IP”页面，单击IP地址后面的“解绑”，在弹出框中单击“是”，解绑EIP。
- 步骤6** 在“弹性公网IP”页面，查看结果。

您也可以在“任务中心”页面，查看解绑弹性公网IP任务的执行进度及结果。

如需重新绑定，请参见[绑定弹性公网IP](#)。

---结束

7.6.3 修改数据库端口

操作场景

关系型数据库服务支持修改主实例和只读实例的数据库端口，对于主备实例，修改主实例的数据库端口，该实例下备实例的数据库端口会被同步修改。

操作步骤

- 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 > 关系型数据库”。进入关系型数据库信息页面。
- 步骤4** 在“实例管理”页面，选择指定的实例，单击主实例名称，或单击 ，单击只读实例名称。
- 步骤5** 在“基本信息”页面，在“连接信息”模块“数据库端口”处，单击 ，修改数据库端口。

说明

SQL Server实例的端口设置范围为1433（默认）或2100~9500（其中5355和5985不可设置）。

- 单击 ，提交修改。
 - 在弹出框中，单击“是”，提交修改。
 - 修改主实例数据库端口，对应的主备实例均会被修改且重启。
 - 修改只读实例数据库端口，仅修改并重启该只读实例。

说明

RDS for SQL Server仅2017企业版支持只读实例，其他版本均不支持只读实例。

- 此过程需要1~5分钟左右。
 - 在弹出框中，单击“取消”，取消本次修改。

步骤6 在实例的“基本信息”页面，查看修改结果。

----结束

7.7 数据安全性

7.7.1 重置管理员密码

操作场景

关系型数据库服务仅支持通过主实例重置管理员密码。

在使用RDS过程中，如果忘记数据库账号密码，可以重新设置密码。

📖 说明

- 当您修改数据库主实例的密码时，如果该实例中存在备实例或只读实例，则会被同步修改。
- 重置密码生效时间取决于该主实例当前执行的业务数据量。
- 请定期（如三个月或六个月）修改用户密码，以提高系统安全性，防止出现密码被暴力破解等安全风险。

方式一

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的实例，选择“更多 > 重置密码”。

步骤5 在“重置密码”弹框，输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*-_+?,特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

方式二

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 > 关系型数据库”。进入关系型数据库信息页面。

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”页签，在“数据库信息”模块的“管理员账户名”处，单击“重置密码”，输入新管理员密码及确认密码。

须知

请妥善保管您的密码，因为系统将无法获取您的密码信息。

所设置的密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%&^*_+?,特殊字符。请您输入高强度密码并定期修改，以提高安全性，防止出现密码被暴力破解等安全风险。

- 单击“确定”，提交重置。
- 单击“取消”，取消本次重置。

----结束

7.7.2 修改实例安全组

操作场景

关系型数据库服务支持修改数据库引擎的主实例和只读实例的安全组，对于主备实例，修改主实例的安全组，备实例的安全组会被同步修改。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择对应的主实例或只读实例，单击实例名称。

步骤5 在“基本信息”页签“连接信息”模块的“安全组”处，单击，选择对应的安全组。

- 单击，提交修改。
- 单击，取消修改。

步骤6 稍后单击“基本信息”页面右上角的，查看修改结果。此过程需1~3分钟。

----结束

7.8 与告警监控指标与告警

7.8.1 支持的监控指标

声明：RDS实例中的Agent只用于监控实例运行的指标、状态，不会收集除监控指标外的其它数据。

功能说明

本节定义了关系型数据库上报云监控的监控指标的命名空间，监控指标列表和维度定义。用户可以通过云监控提供的API接口来检索关系型数据库产生的监控指标和告警信息。

命名空间

SYS.RDS

实例监控指标

- 弹性云服务器相关指标，如表7-4所示。

表 7-4 弹性云服务器性能监控列表

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds001_cpu_util	CPU使用率	该指标用于统计测量对象的CPU利用率。	0~100%	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds003_iops	IOPS	该指标用于统计当前实例，单位时间内系统处理的I/O请求数量（平均值）。	≥0counts/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds039_disk_util	磁盘利用率	该指标用于统计测量对象的磁盘利用率。	0~100%	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds002_mem_util	内存使用率	该指标用于统计测量对象的内存利用率。	0~100%	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期 (原始指标)
rds004_bytes_in	网络输入吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输入的流量。	≥0byte s/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds005_bytes_out	网络输出吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输出的流量。	≥0byte s/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds049_disk_read_throughput	硬盘读吞吐量	该指标用于统计每秒从磁盘读取的字节数。	≥0byte s/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds050_disk_write_throughput	硬盘写吞吐量	该指标用于统计每秒写入磁盘的字节数。	≥0byte s/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds047_disk_total_size	磁盘总大小	该指标用于统计测量对象的磁盘总大小。	40GB ~ 4000GB	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds048_disk_used_size	磁盘使用量	该指标用于统计测量对象的磁盘使用大小。	0GB ~ 4000GB	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds053_avg_disk_queue_length	磁盘平均队列长度	该指标用于统计等待写入测量对象的进程个数。	≥0	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds054_d b_connections_in_use	使用中的数据库连接数	用户连接到数据库的连接数量。	≥0 counts	测量对象：数据库 监控实例类型：Microsoft SQL Server实例	1分钟

- Microsoft SQL Server数据库性能监控指标，如表7-5所示。

表 7-5 数据库性能监控列表

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds001_cpu_util	CPU使用率	该指标用于统计测量对象的CPU使用率，以比率为单位。	0-100 %	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds002_mem_util	内存使用率	该指标用于统计测量对象的内存使用率，以比率为单位。	0-1	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds003_iops	IOPS	该指标用于统计当前实例，单位时间内系统处理的I/O请求数量（平均值）。	≥ 0 count s/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds004_bytes_in	网络输入吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输入的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds005_byte_s_out	网络输出吞吐量	该指标用于统计平均每秒从测量对象的所有网络适配器输出的流量，以字节/秒为单位。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds039_disk_util	磁盘利用率	该指标用于统计测量对象的磁盘利用率，以比率为单位。	0-1	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds047_disk_total_size	磁盘总大小	该指标用于统计测量对象的磁盘总大小。	40GB~4000GB	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds048_disk_used_size	磁盘使用量	该指标用于统计测量对象的磁盘使用大小。	0GB~4000GB	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds049_disk_read_throughput	硬盘读吞吐量	该指标用于统计每秒从硬盘读取的字节数。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds050_disk_write_throughput	硬盘写吞吐量	该指标用于统计每秒写入硬盘的字节数。	≥ 0 bytes/s	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟
rds053_avg_disk_queue_length	磁盘平均队列长度	该指标用于统计等待写入测量对象的进程个数。	≥ 0	测量对象：弹性云服务器 监控实例类型：Microsoft SQL Server实例	1分钟

指标	指标名称	含义	取值范围	测量对象和监控对象	监控周期（原始指标）
rds054_db_connections_in_use	使用中的数据库连接数	用户连接到数据库的连接数量。	≥0 count s	测量对象：数据库 监控实例类型：Microsoft SQL Server实例	1分钟

维度

Key	Value
rds_instance_sqlserver_id	Microsoft SQL Server实例节点ID

7.8.2 设置告警规则

操作场景

通过在云监控中[设置告警规则](#)，用户可自定义关系型数据库的监控目标与通知策略，及时了解关系型数据库运行状况，从而起到预警作用。

设置关系型数据库的告警规则包括设置告警规则名称、服务、维度、监控对象、监控指标、告警阈值、监控周期和是否发送通知等参数。

设置告警规则

- 步骤1** 登录管理控制台。
- 步骤2** 选择“管理与监管 > 云监控服务 CES”。
- 步骤3** 在左侧导航树栏，选择“告警 > 告警规则”。
- 步骤4** 在“告警规则”界面，单击“创建告警规则”进行添加。

----结束

7.8.3 查看监控指标

操作场景

云服务平台提供的云监控，可以对关系型数据库的运行状态进行日常监控。您可以通过管理控制台，直观地查看关系型数据库的各项监控指标。

由于监控数据的获取与传输会花费一定时间，因此，云监控显示的是当前时间5~10分钟前的关系型数据库状态。如果您的关系型数据库刚创建完成，请等待5~10分钟后查看监控数据。

前提条件

- 关系型数据库正常运行。
故障、删除状态的关系型数据库，无法在云监控中查看其监控指标。当关系型数据库再次启动或恢复后，即可正常查看。

📖 说明

故障24小时及以上的关系型数据库，云监控将默认该关系型数据库不存在，并在监控列表中删除，不再对其进行监控，但告警规则需要用户手动清理。

- 关系型数据库已正常运行一段时间（约10分钟）。
对于新创建的关系型数据库，需要等待一段时间，才能查看上报的监控数据和监控视图。

操作步骤

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择目标实例，单击操作列中的“查看监控指标”，跳转到云监控服务页面。

您也可以在“实例管理”页面，单击目标实例名称，在页面右上角，单击“查看监控指标”，跳转到云监控服务页面。

步骤5 在云监控页面，可以查看实例监控信息。

云监控支持的性能指标监控时间窗包括：近1小时、近3小时、近12小时、近24小时、近7天。

----结束

7.9 CTS 审计

7.9.1 支持审计的关键操作列表

通过，您可以记录与关系型数据库实例相关的操作事件，便于日后的查询、审计和回溯。

表 7-6 支持的 RDS 操作列表

操作名称	资源类型	事件名称
创建实例、创建只读、恢复到新实例（Console、OPENAPI、TROVEAPI）	instance	createInstance
扩容、规格变更（Console、OPENAPI、TROVEAPI）	instance	instanceAction
实例重启（Console、OPENAPI、TROVEAPI）	instance	instanceRestart
恢复到原有实例（Console、OPENAPI、TROVEAPI）	instance	instanceRestore
实例重命名（Console）	instance	instanceRename
重置密码（Console）	instance	resetPassword
设置数据库版本配置参数（OPENAPI）	instance	setDBParameters
重置实例的数据库版本配置参数（OPENAPI）	instance	resetDBParameters
设置备份策略-打开，关闭，修改（Console、OPENAPI）	instance	setBackupPolicy
修改数据库端口号（Console）	instance	changeInstancePort
绑定解绑EIP（Console）	instance	setOrResetPublicIP
修改安全组（Console）	instance	modifySecurityGroup
创建标签（Console、OPENAPI）	instance	createTag
删除标签（Console、OPENAPI）	instance	deleteTag
修改标签（Console、OPENAPI）	instance	modifyTag
删除实例（Console、OPENAPI、TROVEAPI）	instance	deleteInstance
开启SQL ServerTDE功能（Console）	instance	sqlserverOpenTDE
主备倒换（Console）	instance	instanceFailOver
修改主备同步模式（Console）	instance	instanceFailOverMode
修改主备倒换策略（Console）	instance	instanceFailOverStrategy
单机转主备实例（Console、OPENAPI、TROVEAPI）	instance	modifySingleToHaInstance

操作名称	资源类型	事件名称
创建快照 (Console、OPENAPI)	backup	createManualSnapshot
复制快照 (Console)	backup	copySnapshot
下载快照 (Console、OPENAPI)	backup	downloadSnapshot
删除快照 (Console、OPENAPI)	backup	deleteManualSnapshot
创建参数模板 (Console、TROVEAPI)	parameterGroup	createParameterGroup
修改参数模板 (Console、TROVEAPI)	parameterGroup	updateParameterGroup
删除参数模板 (Console、TROVEAPI)	parameterGroup	deleteParameterGroup
复制参数模板 (Console)	parameterGroup	copyParameterGroup
重置参数模板 (Console)	parameterGroup	resetParameterGroup
比较参数模板 (Console)	parameterGroup	compareParameterGroup
应用参数模板 (Console)	parameterGroup	applyParameterGroup
保存参数模板 (Console)	parameterGroup	saveParameterGroup
冻结删除 (Console)	all	deleteInstance
实例冻结 (Console)	all	rdsfreezeInstance

7.9.2 查看追踪事件

操作场景

在您开通了云审计服务后，系统开始记录云服务资源的操作。云审计服务管理控制台保存最近7天的操作记录。

本节介绍如何在管理控制台查看最近7天的操作记录。

操作步骤

- 步骤1** 登录管理控制台。
- 步骤2** 选择“管理与监管 > 云审计服务 CTS”，进入云审计服务信息页面。
- 步骤3** 单击左侧导航树的“事件列表”，进入事件列表信息页面。
- 步骤4** 事件列表支持通过筛选来查询对应的操作事件。详细信息如下：

- 事件来源、资源类型和筛选类型：在下拉框中选择查询条件。
其中筛选类型选择资源ID时，还需选择或者手动输入某个具体的资源ID。
- 操作用户：在下拉框中选择某一具体的操作用户。
- 事件级别：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
- 时间范围：可通过选择时间段查询操作事件。

步骤5 选择查询条件后，单击“查询”。

步骤6 在需要查看的记录左侧，单击展开该记录的详细信息。

步骤7 在需要查看的记录右侧，单击“查看事件”，在弹出框中显示该操作事件结构的详细信息。

步骤8 单击右侧的“导出”，将查询结果以CSV格式的文件导出，该CSV文件包含了云审计服务记录的七天以内的操作事件的所有信息。

关于事件结构的关键字段详解，请参见《云审计服务用户指南》的“事件结构”和“事件样例”章节。

----结束

7.10 任务中心

7.10.1 查看任务

您可以通过“任务中心”查看任务执行进度和结果，并进行管理。

说明

关系型数据库服务支持查看和管理以下任务：

- 创建实例。
- 创建只读实例。
- 单实例转主备。
- 磁盘扩容。
- 实例绑定弹性公网IP。
- 实例解绑弹性公网IP。
- 实例主备倒换。
- 重启实例。
- 恢复到新实例。

查看即时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“即时任务”页签下，选择目标任务，查看任务信息。

- 您可以通过任务名称和实例名称/ID确定目标任务，或通过右上角的搜索框输入任务名称来确定目标任务。
- 您还可以单击页面右上角的，查看某一段时间内的任务执行进度和状态，默认时长为7天。
任务保留时长最多为30天。
- 系统支持查看以下状态的即时任务：
 - 执行中
 - 完成
 - 失败
- 查看任务创建时间和结束时间。

----结束

查看定时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“定时任务”页签下，查看目标任务的执行进度和结果。

- 您可以通过实例名称/ID确定目标任务，或通过右上角的搜索框输入实例ID来确定目标任务。
- 系统支持查看以下状态的定时任务：
 - 执行中
 - 完成
 - 失败
 - 取消
 - 待执行
 - 待授权

----结束

7.10.2 删除任务

对于不再需要展示的任务，您可以通过“任务中心”进行任务记录的删除。删除任务仅删除记录，不会删除数据库实例或者停止正在执行中的任务。

须知

删除任务将无法恢复，请谨慎操作。

删除即时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“即时任务”页签下，选择目标任务，单击操作列的“删除”，在弹出框中单击“是”，删除任务。

关系型数据库服务支持删除以下状态的即时任务：

- 执行中
- 完成
- 失败

----结束

删除定时任务

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“任务中心”页面，在“定时任务”页签下，选择目标任务，查看目标任务的任务状态，是否是“待执行”或“待授权”状态。

- 是，请执行**步骤5**。
- 否，请执行**步骤6**。

步骤5 单击操作列的“取消”，在弹出框中单击“是”，取消任务。再次单击操作列的“删除”，在弹出框中单击“是”，删除任务。

步骤6 单击操作列的“删除”，在弹出框中单击“是”，删除任务。

关系型数据库服务支持删除以下状态的定时任务：

- 执行中
- 完成
- 失败
- 取消
- 待执行
- 待授权

----结束

7.11 存储过程的使用

7.11.1 创建主账号

操作场景

使用该存储过程可以创建出一个新的登录账号，该账号拥有SQL Server主帐号rdsuser的所有权限。

说明

- 存储过程只允许使用rdsuser或者创建出的主账号执行，创建的主账号具有和rdsuser相同的权限。
- 密码长度为8~32个字符，至少包含大写字母、小写字母、数字、特殊字符三种字符的组合，其中允许输入~!@#%^*-_+?,特殊字符。

前提条件

成功连接RDS for SQL Server实例。

操作步骤

执行以下命令，创建主账号。执行成功后可以使用新的账号登录验证。

```
EXEC master.dbo.rds_create_major_login @login='loginName',  
@password='password';
```

- loginName：创建的主账号的登录名。
- password：创建的主账号的密码。

示例

创建名为“rdsuser1”，密码为“*****”的账号，示例如下：

```
EXEC master.dbo.rds_create_major_login @login='rdsuser1', @password='*****';
```

7.11.2 更新警报和作业所用的操作员信息

操作场景

使用存储过程更新警报和作业所用的操作员（通知收件人）信息。

前提条件

成功连接RDS for SQL Server实例。通过SQL Server客户端连接目标实例。

操作步骤

执行以下命令，更新警报和作业所用的操作员信息。

```
EXEC [msdb].[dbo].[rds_update_operator]  
  
@name = 'name',  
  
@new_name = 'new_name',
```

@enabled=enabled,
@email_address='email_address',
@pager_address= 'pager_number',
@weekday_pager_start_time= weekday_pager_start_time,
@weekday_pager_end_time= weekday_pager_end_time,
@saturday_pager_start_time= saturday_pager_start_time,
@saturday_pager_end_time= saturday_pager_end_time,
@sunday_pager_start_time= sunday_pager_start_time,
@sunday_pager_end_time= sunday_pager_end_time,
@pager_days= pager_days,
@netsend_address = 'netsend_address',
@category_name='category'

表 7-7 参数说明

参数	说明
'name'	操作员（通知收件人）的名称。此名称必须唯一，并且不能包含百分号（%）字符。名称为sysname，无默认值。
'new_name'	操作员的新名称。此名称必须唯一。new_name是sysname，默认值为NULL。
enabled	指示操作员的当前状态。enabled为tinyint，默认值为1（已启用）。如果为0，则不启用操作员且不会收到通知。
'email_address'	操作员的电子邮件地址。此字符串将直接传递到电子邮件系统。email_address为nvarchar（100），默认值为 NULL。
'pager_number'	操作员的寻呼地址。此字符串将直接传递到电子邮件系统。pager_number的数据类型为 nvarchar(100)，默认值为 NULL。
weekday_pager_start_time	时间，在该时间之后 SQL Server代理将寻呼通知发送到工作日的指定操作员，从星期一到星期五。 weekday_pager_start_time的值为int，默认值为090000，指示9:00 A.M。并且必须使用HHMMSS格式输入。
weekday_pager_end_time	一个时间，在此时间之后，SQLServerAgent服务不再向星期一到星期五的工作日发送寻呼通知。 “weekday_pager_end_time”的值为“int”，默认值为180000，表示6:00 P.M.并且必须使用HHMMSS格式输入。
saturday_pager_start_time	SQLServerAgent服务在星期六向指定操作员发送寻呼通知的时间。“saturday_pager_start_time”的值为“int”，默认值为090000，指示9:00 A.M。并且必须使用HHMMSS格式输入。

参数	说明
saturday_pager_end_time	一个时间，在此时间之后，SQLServerAgent服务不再向周六的指定操作员发送寻呼通知。“saturday_pager_end_time”的值为“int”，默认值为180000，表示6:00 P.M.并且必须使用HHMMSS格式输入。
sunday_pager_start_time	SQLServerAgent服务在星期日向指定操作员发送寻呼通知的时间。“sunday_pager_start_time”的值为“int”，默认值为090000，指示9:00 A.M.并且必须使用HHMMSS格式输入。
sunday_pager_end_time	在星期日上，SQLServerAgent服务不再向指定操作员发送寻呼通知的时间。“sunday_pager_end_time”的值为“int”，默认值为180000，表示6:00 P.M.并且必须使用HHMMSS格式输入。
pager_days	是一个数字，指示操作员可用于页面的日期（受限于指定的开始/结束时间）。pager_days为tinyint，默认值为0，表示运算符从不可用于接收页面。有效值为0至127。pager_days是通过添加所需日期的各个值来计算的。例如，从星期一到星期五是 $2+4+8+16+32 = 62$ 。下表列出了一周中每天的值。 <ul style="list-style-type: none">• 值为1，表示星期日。• 值为2，表示星期一。• 值为4，表示星期二。• 值为8，表示星期三。• 值为16，表示星期四。• 值为32，表示星期五。• 值为64，表示星期六。
'netsend_address'	要向其发送网络消息的操作员的网络地址。netsend_address为nvarchar (100)，默认值为NULL。
'category'	此操作员的类别名称。category的类型为sysname，默认值为NULL。

执行成功后，系统将会如下提示：

```
Commands completed successfully.
```

7.11.3 删除警报

操作场景

使用存储过程删除警报。

前提条件

成功连接RDS for SQL Server实例。通过SQL Server客户端连接目标实例。

操作步骤

执行以下命令，删除警报。

```
EXEC [msdb].[dbo].[rds_delete_alert]  
  
@name='name'
```

表 7-8 参数说明

参数	说明
'name'	警报的名称。名称是sysname，无默认值。

执行成功后，系统将会如下提示：

```
Commands completed successfully.
```

示例

```
EXEC [msdb].[dbo].[rds_delete_alert]  
    @name='test'
```

回显如下所示：

```
Messages  
Commands completed successfully.
```

7.11.4 删除特定警报和操作员的 SQL Server 代理通知定义

操作场景

使用存储过程删除特定警报和操作员的SQL Server代理通知定义。

前提条件

成功连接RDS for SQL Server实例。通过SQL Server客户端连接目标实例。

操作步骤

执行以下命令，删除特定警报和操作员的SQL Server代理通知定义。

```
EXEC [msdb].[dbo].[rds_delete_notification]  
  
@alert_name = 'alert',  
  
@operator_name = 'operator';
```

表 7-9 参数说明

参数	说明
'alert'	警报的名称。警报是sysname，无默认值。

参数	说明
'operator'	操作员的名称。运算符是sysname，无默认值。

执行成功后，系统将会如下提示：

```
Commands completed successfully.
```

示例

```
EXEC [msdb].[dbo].[rds_delete_notification]
    @alert_name = 'alert',
    @operator_name = N'TestOperator';
```

回显如下所示：

```
Messages
Commands completed successfully.
```

7.11.5 删除操作员

操作场景

使用存储过程删除一位操作员。

前提条件

成功连接RDS for SQL Server实例。通过SQL Server客户端连接目标实例。

操作步骤

执行以下命令，删除一位操作员。

```
EXEC [msdb].[dbo].[rds_delete_operator]
    @name='name'
    @reassign_to_operator = 'reassign_operator';
```

表 7-10 参数说明

参数	说明
'name'	要删除的操作员的名称。名称是sysname，无默认值。
'reassign_operator'	可以向其重新分配指定的操作员的警报的操作员的名称。reassign_operator是sysname，默认值为 NULL。

执行成功后，系统将会如下提示：

Commands completed successfully.

示例

```
EXEC [msdb].[dbo].[rds_delete_operator]  
    @name = N'HWTest01';
```

回显如下所示：

```
Messages  
Commands completed successfully.
```

8 常见问题

8.1 产品咨询

8.1.1 使用 RDS 要注意些什么

1. 实例的操作系统，对用户都不可见，这意味着，只允许用户应用程序访问数据库对应的IP地址和端口。
2. 上的备份文件以及关系型数据库服务使用的弹性云服务器（Elastic Cloud Server，简称ECS），都对用户不可见，它们只对关系型数据库服务的后台管理系统可见。
3. 查看实例列表时请确保与购买实例选择的区域一致。
4. 申请RDS后，您还需要做什么。
申请关系型数据库实例后，您不需要进行数据库的基础运维（比如高可用、安全补丁等），但是您还需要重点关注以下事情：
 - a. 关系型数据库实例的CPU、IOPS、空间是否足够，如果不够需要变更规格或者扩容。
 - b. 关系型数据库实例是否存在性能问题，是否有大量的慢SQL，SQL语句是否需要优化，是否有多余的索引或者缺失的索引等。

8.1.2 什么是 RDS 实例可用性

关系型数据库实例可用性的计算公式：

实例可用性=（1-故障时间/服务总时间）×100%

8.1.3 可以通过创建模板的方式创建实例吗

目前不支持实例模板。

8.1.4 RDS 与其他数据库解决方案间的差异

表 8-1 RDS 与其他数据库解决方案间的差异

功能	RDS	自购服务器搭建数据库服务
服务可用性	请参见《弹性云服务器用户指南》。	需自行保障，自行搭建主从复制，自建RAID等。
数据可靠性	请参见《云硬盘用户指南》。	需自行保障，自行搭建主从复制，自建RAID等。
系统安全性	防DDoS，及时修复各种数据库安全漏洞。	自行部署，价格高昂；自行修复数据库安全漏洞。
数据库备份	自动备份。	自行实现，但需要寻找备份存放空间以及定期验证备份是否可恢复。
软硬件投入	无软硬件投入，按需付费。	数据库服务器成本相对较高。
系统托管	无托管费用。	托管费用比较高。
维护成本	无需运维。	需招聘专职DBA来维护，花费大量人力成本。
部署扩容	即时开通，快速部署，扩容，按需开通。	需硬件采购、机房托管、部署机器等工作，周期较长。
资源利用率	按实际结算，利用率高。	考虑峰值，资源利用率很低。

8.1.5 RDS 实例是否会受其他用户实例的影响

关系型数据库实例不会受其他用户实例影响，因为每个用户的关系型数据库实例与其他用户的实例是相互独立的，并且有资源隔离，互不影响。

8.1.6 关系型数据库支持跨 AZ 高可用吗

RDS支持跨AZ高可用。当用户创建实例的时候，选择主备，可以选择主可用区和备可用区不在同一个可用区（AZ）。

8.1.7 RDS 是否支持主备实例变更为单机实例

RDS仅支持单机实例变更为主备实例，主备实例无法变更为单机实例。

8.1.8 导出 SQL 查询结果到 Excel 出现乱码

编码导致出现乱码，默认是utf8，需要在导出的Excel中，将默认编码转换为Unicode。

8.1.9 如何搭建 AD 域

本章节介绍如何搭建AD（Active Directory）域，并将RDS for SQL Server实例加入域中。

操作步骤

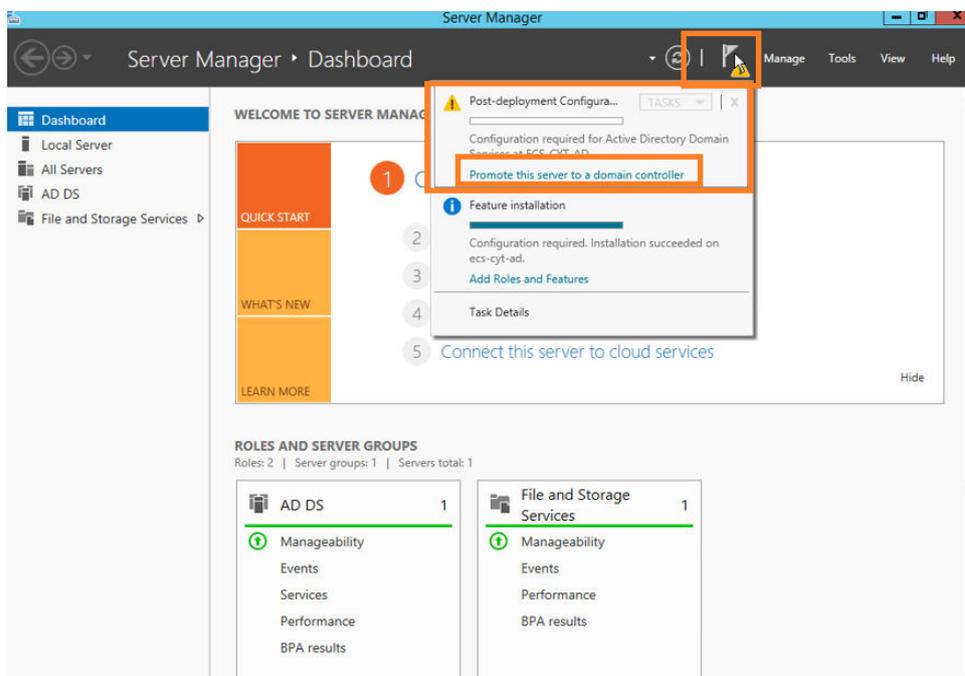
本章使用windows server 2012 r2搭建域服务器。

说明

1. 创建加域实例的时候，用户的域控服务器不能配置域级别的组策略对象（GPO，Group Policy Object），或者域级别GPO要禁用掉，否则用户创建加域实例将会失败。
2. 如果需要应用域级别的GPO，需要另外创建ECS搭建新的域控服务器，并且该域控服务器不能设置应用域级别的GPO，然后创建加入该域的实例，将用户侧域控服务器和新的域控服务器建立信任关系即可使用SQL Server加域实例。

步骤1 安装AD域控制器。

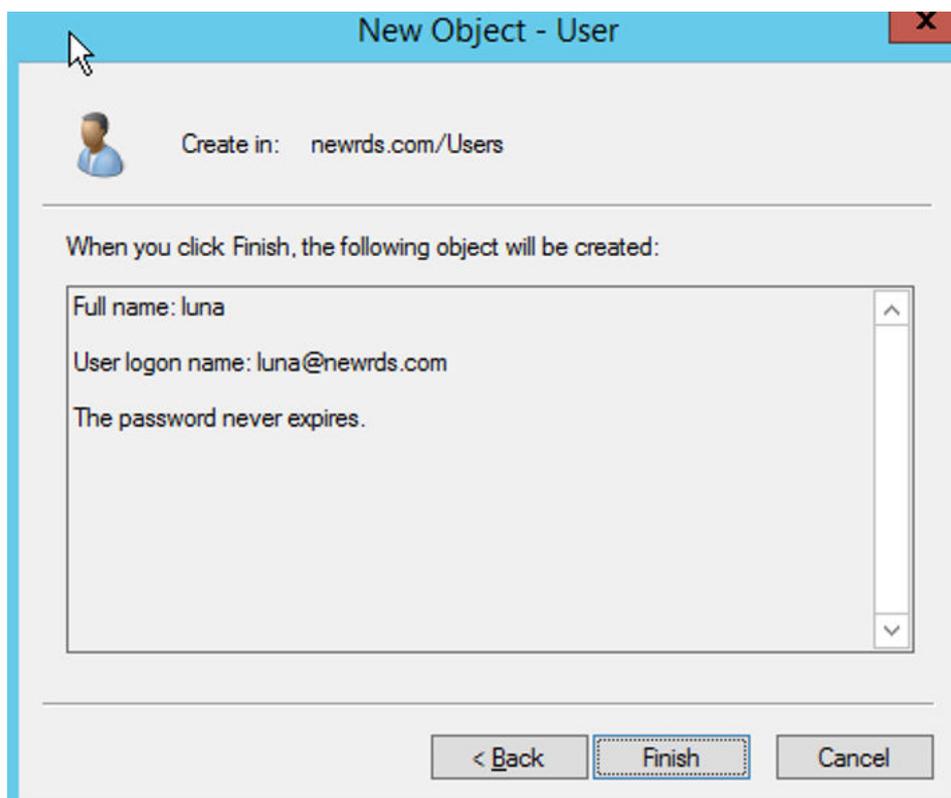
1. 打开服务管理器，单击“添加角色和功能”。
2. 单击“下一步”，直到“选择服务器角色”页面，勾选“Active Directory域服务”选项，并在弹框中单击“确认添加”。
3. 单击“下一步”，直到“安装”页面，单击“安装”，启动角色安装流程。
4. 成功安装之后，如下图所示会有一个黄色三角提示标，单击“将此服务器提升为域服务器”，进入Active Directory域服务配置向导。



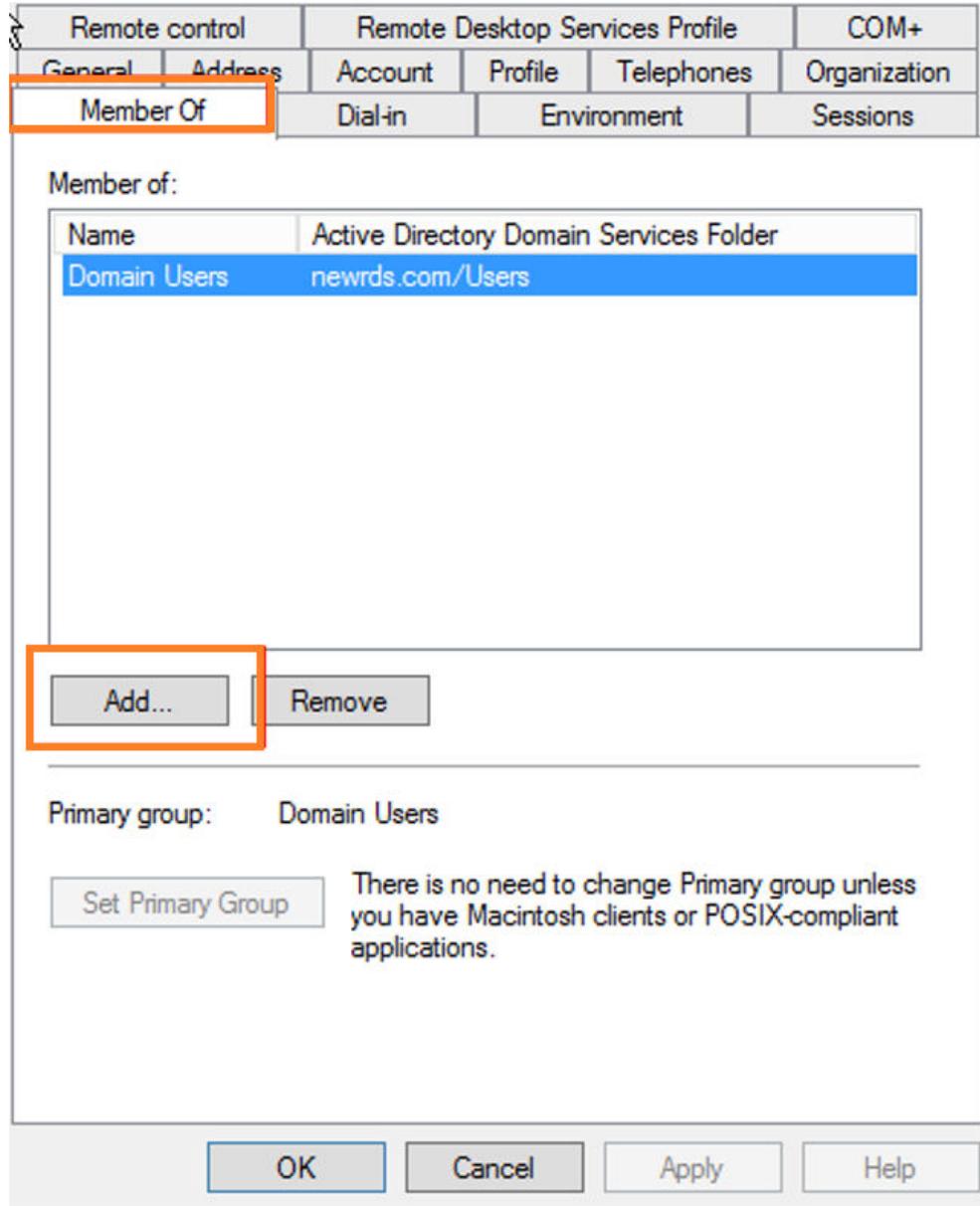
5. 在“部署配置”页面，选择“添加新林”，并设置域名（如：newrds.com）。
6. 单击“下一步”，配置域服务器参数，输入DSRM的密码（非域用户）。
7. 单击“下一步”，直到“安装”页面，单击“安装”，开始安装AD域服务器，安装完成后会自动重启。
8. 修改网络接口中的DNS配置。将主DNS服务器设置为自己的IP地址。（如：192.168.0.133）

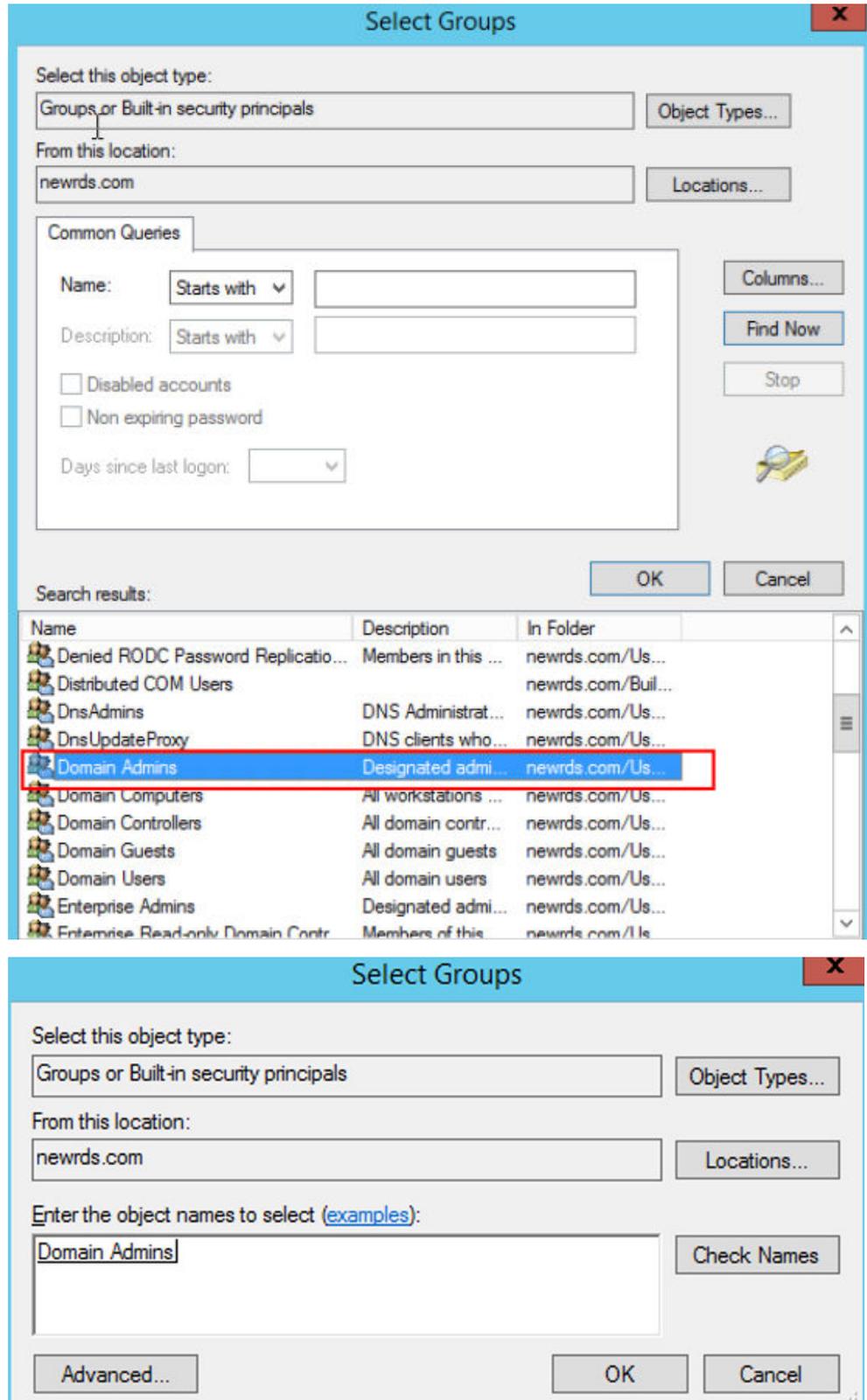
步骤2 创建并添加域账号。

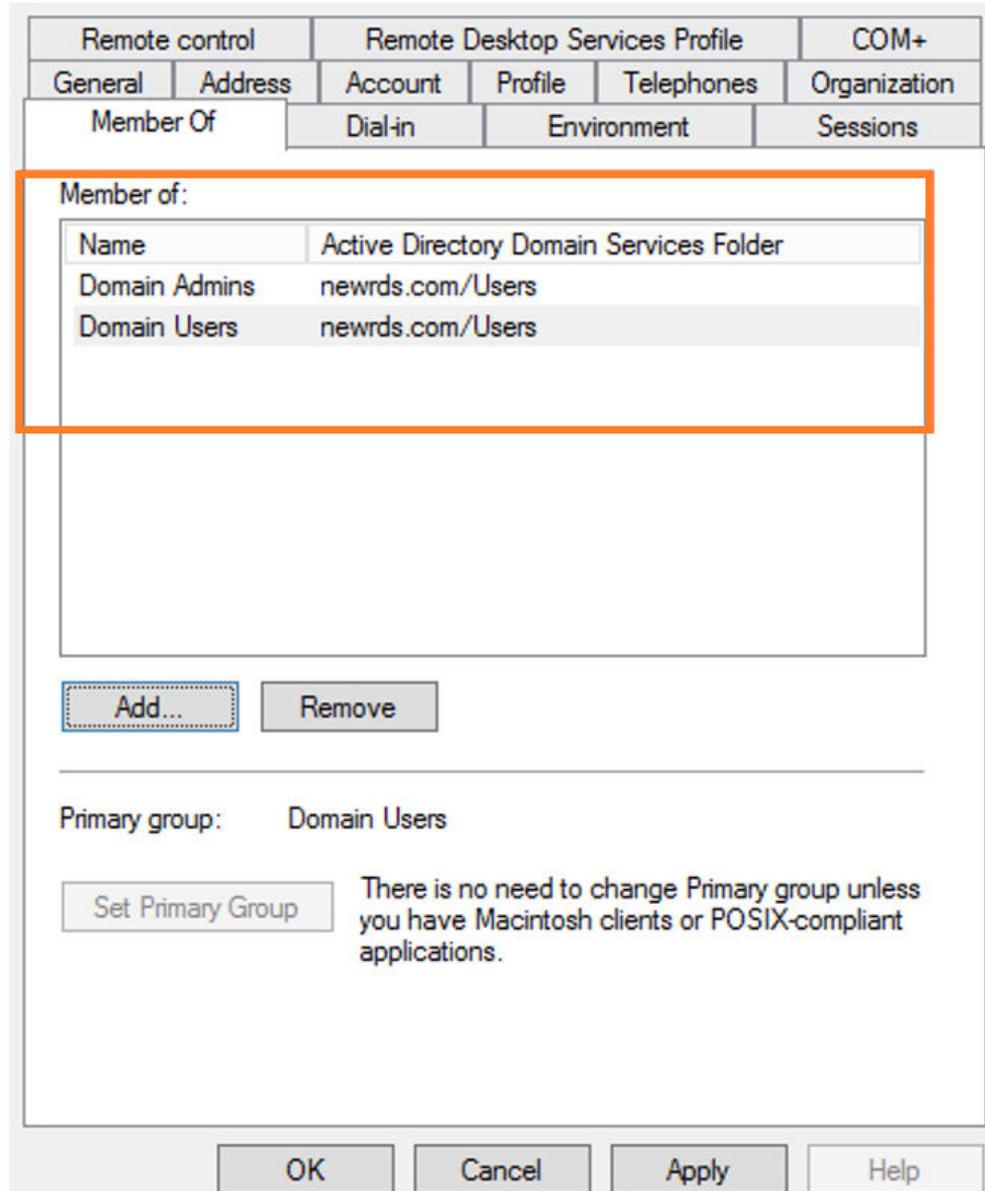
1. 打开“Active Directory用户和计算机”，右键单击“新建”，选择“用户”。输入用户名称，单击“下一步”。
2. 输入“名称”、“姓氏”和“用户登录名”。（如：luna@newrds.com）
3. 填写域账号密码和确认密码。取消所有勾选框的检查（第一次登录时不要更改密码）。
4. 添加成功后如下图所示。可以将域账号添加到用户组中进行权限控制。



5. 对于需要登录RDS实例的域账号，请将用户添加至AD Admin组中。







步骤3 将RDS云实例加入域中。

仅需要在正常实例创建页面，单击“现在配置”。单击后，实例会去进行加域操作，创建成功后即可使用。

表 8-2 AD 域参数配置

参数	说明
目录地址	支持Active Directory域服务的弹性云服务器所在IP地址。 示例：192.168.x.x。 说明 请确认该IP地址可以成功连接云数据库RDS实例。

参数	说明
域名	一个完全限定域名，示例：DBStest.com，域名有如下限制： 1. 该域名必须和用户设置的弹性云服务器的域名保持一致。 2. 域名长度不超过48个字符。 3. 域名只能包含：字母、数字、中划线和英文句点。 4. 域名必须包含扩展名，且扩展名长度大于等于2个字符，不能包含数字，例如.com。
域账号名称	建议设置为域管理员账号。
域账号密码	您所设置的域账号密码。 请妥善保管您的密码，因为系统将无法获取您的密码信息。

----结束

8.1.10 为何使用了 RDS 后网站登录较慢

推荐您做如下两个处理：

- 通过关系型数据库服务的管理控制台查看关系型数据库实例的性能情况。
- 与应用程序有很大关系，使用命令查看当前数据库连接状态，比较本地数据库和关系型数据库的差异。

8.1.11 云数据库如何进行主备切换

关系型数据库（Relational Database Service，简称RDS）服务提供高可用类型，推荐您选择主备模式。

故障切换

也叫计划外的切换。当主机出现故障时，系统会在1~5分钟内自动切换到备机，主备实例的连接IP不变，整个过程无需人工干预。切换过程中不可访问，需要您设置好程序跟关系型数据库服务的自动重连，避免因切换导致服务不可用。

手动切换

也叫计划内的切换。当实例运行正常时，用户可以自主手动触发主备切换，以满足业务需求。

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 > 关系型数据库”。进入关系型数据库信息页面。

步骤4 在“实例管理”页面，选择指定的主备实例，单击实例名称，进入实例的“基本信息”页面。

步骤5 在“基本信息”页面中“数据库信息”模块的“实例类型”处，单击“主备切换”。

您也可以在“基本信息”页面，单击“实例拓扑图”模块的 。进行主备切换。

须知

主备切换可能会造成几秒或几分钟的服务闪断（闪断时间与复制时延有关），并有可能在主备同步时延过大的情况下，导致少量数据丢失。主备切换后，请注意对业务进行预热，避免业务高峰期出现阻塞。

步骤6 在“主备切换”弹框，单击“是”进行主备实例的切换。

在“复制状态”为“正常”的情况下，复制时延大于300s，主备切换任务无法下发。

步骤7 主备切换成功后，单击“返回实例列表”，用户可以在“实例管理”页面对其进行查看和管理。

- 切换过程中，状态显示为“主备切换中”。
- 在实例列表的右上角，单击  刷新列表，可查看到主备切换完成后，实例状态显示为“正常”。

----结束

8.1.12 多台弹性云服务器是否可以使用同一个 RDS 数据库

在数据库的压力承载范围内，多台弹性云服务器是可以使用同一个关系型数据库来支撑业务的。

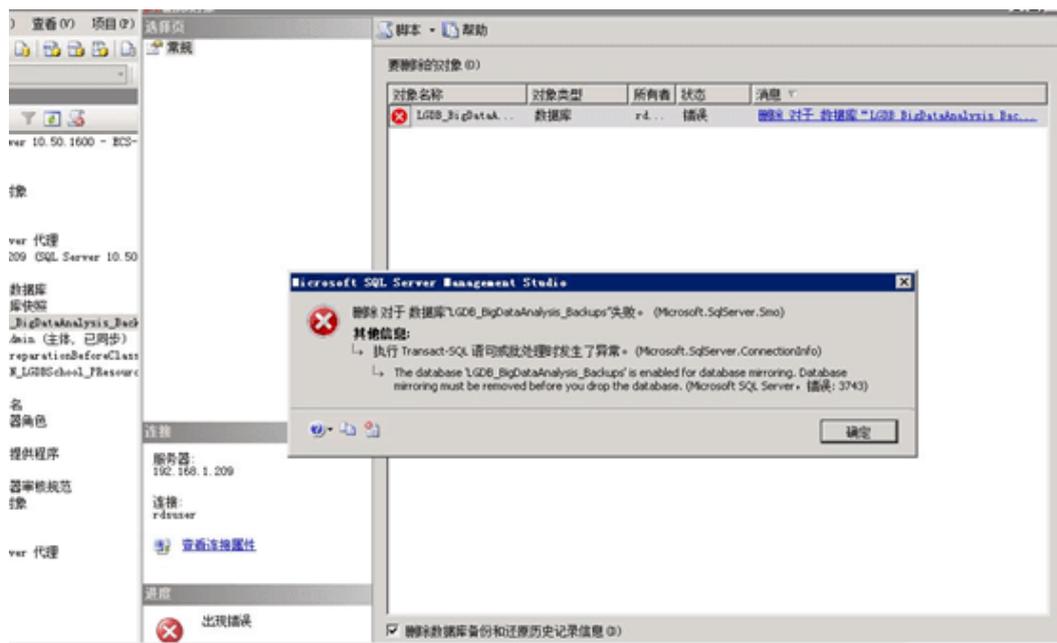
8.1.13 删除 RDS SQL Server 主备实例数据库报错

报错现象

手动删除RDS SQL Server主备实例上的sqlserver数据库，SSMS提示如下报错信息，详见图8-1：

“The database ‘xxxx’ is enabled for database mirroring.Database mirroring must be removed before you drop the database.错误：3743”

图 8-1 报错信息



原因分析

根据报错信息判断，该SQL Server实例为主备模式，备库存在镜像，导致主库无法删除。

解决方案

您需要在删除数据库之前，执行以下命令来禁用镜像：

```
Use master
```

```
go
```

```
ALTER DATABASE [Database_Name] SET PARTNER OFF;
```

```
GO
```

数据库镜像删除后，您就可以删除数据库。

8.2 资源及磁盘管理

8.2.1 占用 RDS 磁盘空间的日志及文件有哪些

占用关系型数据库实例的磁盘空间的日志及文件如下表：

表 8-3 MySQL 数据库文件类型

数据库引擎	文件类型
MySQL	日志文件：数据库undo-log、redo-log和Binlog文件。
	数据文件：数据库内容文件和索引文件。
	其他文件：ibdata、ib_logfile0和临时文件等。

表 8-4 PostgreSQL 数据库文件类型

数据库引擎	文件类型
PostgreSQL	日志文件：数据库错误日志文件和事务日志文件。
	数据文件：数据库内容文件、索引文件、复制槽数据文件、事务状态数据文件和数据库配置文件。
	其他文件：临时文件。

表 8-5 Microsoft SQL Server 数据库文件类型

数据库引擎	文件类型
Microsoft SQL Server	日志文件：数据库的错误日志、事务日志文件和跟踪文件。
	数据文件：数据库内容文件。

解决方案

- 随着业务数据的增加，原来申请的数据库磁盘容量可能会不够用，您需要为关系型数据库实例进行扩容。
- 针对数据空间过大，可以删除无用的历史表数据进行释放空间（DROP或TRUNCATE操作，如果是执行DELETE操作，需要使用OPTIMIZE TABLE来释放空间）；如果没有可删除的历史数据，需要进行磁盘扩容。
- 针对大量排序查询导致的临时文件过大，建议进行优化SQL查询。
 - 应用中存在大量的排序查询，产生了大量的临时文件。
 - 短期内大量增、删、改，产生大量binlog文件占用空间。
 - 由于大量的事务和写入操作产生了大量的binlog日志文件。
- 目前可以监控存储空间的大小、使用量、利用率等，并且设置告警策略。

8.2.2 哪些内容会占用用户所创建的 RDS 实例空间

用户正常的数据库数据（不包括备份数据），以及关系型数据库实例正常运行所需的数据（比如系统数据库、数据库回滚日志、重做日志、索引等）。同时，包含如下关系型数据库的必要日志文件：

- MySQL数据库产生的Binlog日志。
- PostgreSQL数据库服务器产生的日志文件。
- Microsoft SQL Server数据库产生的日志文件（Microsoft SQL Server日志、默认的Trace文件和代理日志文件等）。

这些文件保证关系型数据库实例正常稳定地运行。

8.2.3 用户申请的 RDS 实例磁盘空间会有哪些系统开销

您申请的磁盘空间会有必要的文件系统开销，这些开销主要包括inode、reserved block，以及数据库运行必需的空间。

8.2.4 用户必须进行 DDL 操作时对磁盘空间的要求

为了确保业务正常，您应该避免在业务高峰进行DDL（Data Definition Language）等可能导致磁盘空间爆增的操作，必须进行DDL操作时，请务必确保磁盘空间大于等于表空间两倍大小+10GB，即：假设您的表空间为500GB，那么，您在进行DDL操作时，需确保磁盘空间大于等于 $500 \times 2 + 10 = 1010$ GB。

8.2.5 使用 RDS 可以运行多少个 RDS 实例

关系型数据库服务对运行实例数量没有限制。

8.2.6 在一个 RDS 实例内能运行多少个数据库

关系型数据库服务没有强制限制在一个RDS实例内能运行多少个数据库，这取决于数据库引擎自身设置。

如果CPU、内存和磁盘大小均满足的情况下，实例允许数据库数量没有限制。数据库中表的数量会影响备份速度。表数量超过50万，备份将会失败。

- MySQL创建的数据库、表数量没有限制。这取决于MySQL自身，详情请查看MySQL官方文档。
- PostgreSQL可创建无限个数据库、无限个数据库账号。
- Microsoft SQL Server可创建100个数据库、无限个数据库账号。

8.3 数据库连接

8.3.1 外部服务器能否访问 RDS 数据库

开通公网访问的实例

对于开通公网访问功能的关系型数据库实例，可以通过外网进行访问。

未开通公网访问的实例

- 在虚拟私有云中开通虚拟专用网络（Virtual Private Network，简称VPN），通过虚拟专用网络连接关系型数据库。
- 将关系型数据库与弹性云服务器创建在同一个虚拟专用网络下，通过弹性云服务器来访问关系型数据库。

8.3.2 RDS 数据库连接数满的排查思路

数据库连接数表示应用程序可以同时连接到数据库的数量，与您的应用程序或者网站能够支持的最大用户数没有关系。

数据库连接数过多，可能会导致业务侧无法正常连接，也会导致实例全量备份和增量备份失败，影响业务的正常使用。

排查思路

1. 请及时排查业务侧连接是否有效，优化实例连接，释放不必要的连接。
2. 规格偏小，请对数据库进行规格扩容。
3. 云监控服务目前可以监控数据库cpu、内存、磁盘、连接数等指标，并且设置告警策略，出现告警时可以提前识别风险。具体请参考《云监控服务用户指南》。

8.3.3 RDS 数据库实例支持的最大数据连接数是多少

关系型数据库服务对此未做限制，取决于数据库引擎参数的默认值和取值范围，例如MySQL引擎的max_connections和max_user_connections参数，用户可在参数模板自定义。

修改最大连接数

支持命令方式修改最大连接数。

1. 查看最大连接数。
`show global variables like 'max_connections';`
2. 在my.cnf文件中修改mysqld下参数max_connections的值。
`[mysqld]`
`max_connections = 1000`

MySQL 最大连接数取值多少合适

- MySQL允许客户端同时连接的数量除了受制于参数max_connections外，还受操作系统对单进程允许打开最大文件数的限制。例如，当操作系统控制每个进程的最大打开文件数为100时，即使设置参数max_connections为200也不会起作用。查看操作系统对单进程打开最大文件数的限制（默认为1024）。

ulimit -n

MySQL的参数open_files_limit是在MySQL启动过程中从操作系统读取的单进程打开最大文件数的值。

查看参数open_files_limit的值。

`show variables like 'open_files_limit';`

- 取值建议
MySQL的最大连接数可以自定义取值，但前提是实例的规格可以支撑该值，最大连接数与实例的内存规格息息相关，具体请参考[关于max_connections](#)。
在实际业务中，最大连接数并不是设置的越大越好，建立的连接数越多，对应的资源开销也就越多。

关于 max_connections

max_connections: 允许同时连接的客户端总数。如果设定值为default，表示该参数和数据库实例的内存（单位：GB）相关，计算公式如下：

max_connections上限估计数值 = 节点可用内存 / 单个连接预估占用内存

📖 说明

- 节点可用内存=总内存 - Buffer Pool占用内存 - 1GB（mysqld进程/操作系统/监控程序等）
- 单个连接预估占用内存（single_thread_memory）= thread_stack（256KB）+ binlog_cache_size（32KB）+ join_buffer_size（256KB）+ sort_buffer_size（256KB）+ read_buffer_size（128KB）+ read_rnd_buffer_size（256KB）= 大约1MB

不同内存规格配置的默认max_connections值如下表所示。

表 8-6 不同内存规格配置的默认 max_connections 值

内存(GB)	连接数
512	100000
384	80000
256	60000
128	30000

内存(GB)	连接数
64	18000
32	10000
16	5000
8	2500
4	1500
2	800

8.3.4 如何创建和连接 ECS

1. 创建弹性云服务器，请参见《弹性云服务器用户指南》。
 - 如果是通过内网连接RDS实例，该弹性云服务器用于连接关系型数据库实例，需要与目标实例处于同一虚拟私有云内。如果通过公网连接RDS实例，该弹性云服务器可以与目标实例不在同一个虚拟私有云内。
 - 正确配置安全组，使得弹性云服务器可以通过“连接地址”访问关系型数据库实例。
2. 连接弹性云服务器，请参见《弹性云服务器用户指南》中“登录弹性云服务器”的内容。

8.3.5 内网方式下 ECS 无法连接 RDS 实例的原因

遇到该问题，参考以下步骤排查解决。

步骤1 先确认弹性云服务器和关系型数据库实例是否在同一个虚拟私有云下。

- 如果在，执行**步骤2**。
- 如果不在，需要重新创建弹性云服务器实例，使之和关系型数据库实例在同一个虚拟私有云下。

步骤2 查看弹性云服务器实例是否添加安全组。

- 如果有，检查安全组的配置规则是否满足要求，MySQL相关，请参见**步骤一：创建实例**中“安全组”的描述，然后执行**步骤3**。PostgreSQL相关，请参见**步骤一：创建实例**中“安全组”的描述，然后执行**步骤3**。Microsoft SQL Server相关，请参见**步骤一：创建实例**中“安全组”的描述，然后执行**步骤3**。
- 如果没有，从弹性云服务器的实例详情页面，进入虚拟私有云页面，选择“安全组”，添加安全组。

步骤3 在弹性云服务器上，测试是否可以正常连接到关系型数据库实例地址的端口。

MySQL的默认端口为3306。

PostgreSQL的默认端口为5432。

Microsoft SQL Server的默认端口为1433。

```
telnet <连接地址> {端口号}
```

- 如果可以通信，说明网络是正常的。
- 如果端口不通，请联系售后技术支持协助排查。

----结束

8.3.6 客户端问题导致连接失败

客户端问题导致连接关系型数据库失败，可以从以下几个方面检查。

1. 弹性云服务器的安全策略

对于Windows平台，可检查Windows的安全策略是否开放关系型数据库端口。对于Linux平台，可使用iptables检查防火墙及端口的放行情况。

2. 应用配置错误

常见的有连接地址写错、端口参数配置错误和JDBC等的连接参数配置错误。

3. 用户名或密码错误

如果连接数据库时出现类似如下错误，请检查用户名或密码是否正确。

- [Warning] Access denied for user 'username'@'yourIp' (using password: NO)
- [Warning] Access denied for user 'username'@'yourIp' (using password: YES)
- Login failed for user 'username'

说明

如问题仍未解决，请联系售后技术支持。

8.3.7 服务端问题导致连接失败

关系型数据库服务端可能出现的问题如下，请依次进行检测。

1. 连接方式有误。

解决方法：检查连接方式。如果是通过内网连接RDS实例，弹性云服务器与关系型数据库实例必须处于同一虚拟私有云内，且只能通过弹性云服务器连接。如果通过公网连接RDS实例，该弹性云服务器可以与目标实例不在同一个虚拟私有云内。

2. 连接数满。

解决方法：通过关系型数据库服务的资源监控功能查看连接数、CPU使用率等指标是否正常。如果达到上限，需要重启关系型数据库实例数据库，断开实例连接或升级关系型数据库实例规格解决。

3. 实例状态异常。比如实例重启卡住，关系型数据库系统故障，实例或表被锁定等。

解决方法：尝试重启功能。如果无法解决，请联系售后技术支持。

8.3.8 应用程序怎么访问 VPC 中的 RDS 实例

您需要确保您的应用程序所在的弹性云服务器与关系型数据库实例在同一个虚拟私有云。如果不在，需要修改虚拟私有云中的路由表和网络ACL，以确保应用程序所在的弹性云服务器能够访问关系型数据库实例。

8.3.9 应用程序是否需要支持自动重连数据库

建议您的应用程序支持自动重连数据库，当数据库重启后，无需人工介入，应用会自动恢复，提供更高的可用性。

同时建议您的应用程序采用长连接方式连接数据库，以降低资源消耗，提高性能。

8.3.10 如何通过 JDBC 连接 PostgreSQL 数据库

通过JDBC连接实例的方式有无需下载SSL证书连接和用户下载SSL证书连接两种，其中使用SSL证书连接通过了加密功能，具有更高的安全性。

前提条件

用户需要具备以下技能：

- 熟悉计算机基础知识。
- 了解java编程语言。
- 了解JDBC基础知识。

驱动获取及使用

- JDBC驱动下载地址：<https://jdbc.postgresql.org/download/>
- JDBC接口使用指南请参考：<https://jdbc.postgresql.org/documentation/>

使用 SSL 证书连接

📖 说明

该方式属于SSL连接模式，需要下载SSL证书，通过证书校验并连接数据库。

您可以在“实例管理”页面，单击实例名称进入“基本信息”页面，单击“数据库信息”模块“SSL”处的, 下载根证书或捆绑包。

步骤1 通过JDBC连接PostgreSQL数据库，代码中的JDBC链接格式如下：

```
jdbc:postgresql://<instance_ip>:<instance_port>/<database_name>?sslmode=verify-full&sslrootcert=<ca.pem>
```

表 8-7 参数说明

参数	说明
<instance_ip>	如果通过弹性云服务器连接，“instance_ip”是主机IP，即“基本信息”页面该实例的“内网地址”。 如果通过连接了公网的设备访问，“instance_ip”为该实例已绑定的“弹性公网IP”。
<instance_port>	端口，默认5432，当前端口，参考“基本信息”页面该实例的“数据库端口”。
<database_name>	数据库名，即需要连接的数据库名（默认的管理数据库是postgres）。
sslmode	ssl连接模式，默认全认证模式。

参数	说明
sslrootcert	ssl连接CA证书路径，该文件需放在执行该命令的路径下。

连接PostgreSQL数据库的java代码，可参考以下示例：

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class MyConnTest {
    final public static void main(String[] args) {
        Connection conn = null;
        // set sslmode here.
        // with ssl certificate and path.
        String url = "jdbc:postgresql://192.168.0.225:5432/my_db_test?sslmode=verify-
full&sslrootcert=/home/Ruby/ca.pem";

        try {
            Class.forName("org.postgresql.Driver");
            conn = DriverManager.getConnection(url, "root", "password");
            System.out.println("Database connected");

            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM mytable WHERE columnfoo = 500");
            while (rs.next()) {
                System.out.println(rs.getString(1));
            }

            rs.close();
            stmt.close();
            conn.close();
        } catch (Exception e) {
            e.printStackTrace();
            System.out.println("Test failed");
        } finally {
            // release resource ....
        }
    }
}
```

----结束

无证书连接

说明

该方式属于SSL连接模式，但不与服务端进行证书校验，用户无需下载SSL证书。

步骤1 通过JDBC连接RDS PostgreSQL数据库实例，代码中的JDBC链接格式如下：

```
jdbc:postgresql://<instance_ip>:<instance_port>|<database_name>?sslmode=disable
```

表 8-8 参数说明

参数	说明
<instance_ip>	如果通过弹性云服务器连接，“instance_ip”是主机IP，即“基本信息”页面该实例的“内网地址”。
	如果通过连接了公网的设备访问，“instance_ip”为该实例已绑定的“弹性公网IP”。
<instance_port>	端口，默认5432，当前端口，参考“基本信息”页面该实例的“数据库端口”。
<database_name>	数据库名，即需要连接的数据库名（默认的管理数据库是postgres）。
sslmode	ssl连接模式， disable 模式表示不进行数据加密。

连接PostgreSQL数据库的java代码，可参考以下示例：

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class MyConnTest {
    final public static void main(String[] args) {
        Connection conn = null;
        // set sslmode here.
        // no ssl certificate, so do not specify path.
        String url = "jdbc:postgresql://192.168.0.225:5432/my_db_test?sslmode=disable";
        try {
            Class.forName("org.postgresql.Driver");
            conn = DriverManager.getConnection(url, "root", "password");
            System.out.println("Database connected");

            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM mytable WHERE columnfoo = 500");
            while (rs.next()) {
                System.out.println(rs.getString(1));
            }
            rs.close();
            stmt.close();
            conn.close();
        } catch (Exception e) {
            e.printStackTrace();
            System.out.println("Test failed");
        } finally {
            // release resource ....
        }
    }
}
```

----结束

8.3.11 SQL Server 连接不上的判断方法

场景排查

- 弹性云服务器能否ping通关系型数据库实例。
如果ping不通，可以查看弹性云服务器和关系型数据库实例是否处于同一个虚拟私有云内，是否使用同一个安全组。
内网连接方式下，弹性云服务器与关系型数据库实例必须处于同一虚拟私有云内，且只能通过弹性云服务器连接。公网连接方式下，该弹性云服务器可以与目标实例不在同一个虚拟私有云内。
- 检查连接IP和端口号是否正确。
注意IP和端口号之间使用“逗号”隔开。
- 检查关系型数据库服务的运行状态是否异常。
- 查看用户名密码是否正确，尝试重置密码。
- 尝试重启关系型数据库实例，检查是否改善。

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 > 关系型数据库”。进入关系型数据库信息页面。
 - 步骤4** 在“实例管理”页面，单击目标实例名称，进入“基本信息”页面和“备份恢复”页面确定连接信息和备份信息。
 - 步骤5** 在实例的“基本信息”页面查看管理员账户名。
 - 步骤6** 下载并在弹性云服务器上安装SSMS客户端。
 - 步骤7** 从弹性云服务器连接到关系型数据库实例。
- 结束

8.3.12 RDS 跨地域内网能访问吗

跨地域内网默认不能访问，不同区域的云服务之间内网互不相通。您可以通过云连接或者VPN打通网络实现内网访问。

8.3.13 使用 SSL 连接数据库发生主备切换后是否可以正常连接

如果数据库使用SSL连接，此时发生了主备实例切换，SSL证书对于主库和备库仍然有效，可以正常连接。

8.3.14 MySQL 支持 SSL 连接吗

MySQL支持SSL连接，与其他友商不同之处是，MySQL默认在数据库服务器端打开SSL开关，用户使用客户端连接时，可根据需要指定是否需要SSL连接。

有关SSL详细介绍，请参考[通过公网连接MySQL实例](#)。

8.3.15 为什么重置密码后新密码没有生效

可能原因

密码重置发生在备份操作之后，执行恢复操作时，恢复为旧密码。

定位思路

密码重置之后是否执行了恢复操作。

解决方法

在Console页面，重新执行“重置密码”，具体请参见[重置管理员密码](#)。

8.4 数据库迁移

8.4.1 为什么要采用 mysqldump 和 pg_dump 工具迁移

该方案优点是简单，容易上手，缺点是停机时间较长。因此它适用于数据量不大，或者允许停机的时间较长，并且在这个时间范围内能够完成的数据。

由于关系型数据库服务提供的服务与原来的数据库服务是完全兼容的，所以对于用户来说，从原有数据库迁移到关系型数据库的过程，与从一台数据库服务器迁移到另外一台数据库服务器的过程基本类似。

8.4.2 RDS 支持导入哪些数据库引擎的数据

- 相同引擎数据库之间数据导入导出，称之为同构型数据库之间数据导入导出。
- 不同引擎数据库之间数据导入导出，称之为异构型数据库之间数据导入导出。例如，从Oracle导入数据到RDS支持的数据库引擎。

异构型数据库之间由于格式不同，不支持直接导入导出。但只要导入导出的格式数据兼容，理论上，导入表数据也是可行的。

异构型数据库之间数据导入导出，一般需要第三方软件，通过数据复制的方式来实现。比如，先使用工具从Oracle中，以文本的格式导出表记录，然后利用Load语句导入到关系型数据库支持的数据库引擎。

8.5 数据库权限

8.5.1 root 账号为什么没有 super 权限

大部分的关系型数据库云服务平台，都未给root账号提供super权限。因为一旦用户拥有了super权限，就可以执行很多管理性的命令，比如reset master, set global..., kill, reset slave等，这很有可能导致主备关系异常而出现故障。这一点是云服务平台和本地搭建MySQL比较大的区别，RDS提供服务化能力，那就需要保证实例的稳定正常运行。

对于客户要求super权限的场景，RDS可以提供服务化能力，也可以通过其他手段绕过super权限的限制。

举例如下：

1. 举例1：有些用户喜欢登录数据库执行如下命令来修改参数，这在RDS是被禁止的，您只能通过RDS界面中的参数修改功能来实现。

set global 参数名=参数值

如果您的脚本中包含**set global**命令导致super缺失，请删除**set global**命令，通过console的参数修改。

2. 举例2：有些用户执行如下命令报错，这也是因为没有super权限导致的，只需要去除**definer='root'**关键字即可。

create definer='root'@'%' trigger(procedure)...

如果您缺失super权限，可以使用mysqldump导入数据，请参考[使用mysqldump迁移MySQL数据](#)导入和导出数据。

3. 举例3：如果您在创建PostgreSQL插件时缺少super权限，请参考[创建和删除插件](#)进行创建扩展。

8.6 数据库存储

8.6.1 RDS for MySQL 支持哪些存储引擎

数据库存储引擎就是一种数据存储方式。使用数据存储引擎实现**存储、处理和保护数据的核心服务**。利用数据库引擎可控制访问权限并快速处理事务，从而满足企业内大多数需要处理大量数据的应用程序要求。

云数据库MySQL数据库只有InnoDB存储引擎支持完整的备份、恢复等服务功能，因此RDS for MySQL推荐使用InnoDB引擎。

在MySQL 5.6.40、MySQL 5.7.22 以上的版本中，不再支持如下存储引擎，主要原因如下：

RDS for MySQL目前不支持MyISAM引擎，主要原因如下：

- MyISAM引擎表不支持事务，仅支持表级别锁，导致读写操作相互冲突。
- MyISAM对数据完整性的保护存在缺陷，且这些缺陷会导致数据库数据的损坏甚至丢失。
- MyISAM在出现数据损害情况下，很多都需要手动修复，无法通过产品服务提供的恢复功能进行数据恢复。
- MyISAM向InnoDB的迁移透明，大多数情况不需要改动建表的代码，云数据库自动转换InnoDB即可完成迁移。

RDS for MySQL目前不支持FEDERATED引擎，主要原因如下：

- 主备实例支持FEDERATED引擎会导致在远端数据库上相同DML重复执行，导致数据错乱。
- FEDERATED引擎会在时间点恢复场景，当全量恢复完成后，远端数据库上数据不会跟随全量备份恢复到全备时的数据状态，在增量恢复阶段再应用数据会导致FEDERATED表数据错乱。

RDS for MySQL目前不支持Memory引擎，主要原因如下：

- 如果内存表隐式的变空，那在Open表的时候数据库就会自己产生一个DELETE event到binlog中。这样当HA集群使用了内存表，那么重启HA，备库（或者只读

库) 就会自己产生一个自己的GTID, 导致主备不一致, 进而引发备库重建, 甚至导致备库会不停的重建。

- 使用Memory表, 会存在OOM的风险, 导致服务被终止。

8.6.2 RDS 存储的存储配置是什么

关系型数据库存储采用云硬盘, 关于云硬盘具体信息, 请参见《云硬盘用户指南》。

关系型数据库的备份数据存储采用对象存储服务, 不占用用户创建的数据库空间。关于关系型数据库实例存储的硬件配置, 请参见《对象存储服务用户指南》。

8.6.3 RDS 是否支持存储类型变更, 普通 IO 转超高 IO

不支持。当RDS实例已经创建成功后, 不支持存储类型的变更。

表 8-9 RDS 不支持的变更项

变更项	变更项说明
存储类型变更	<ul style="list-style-type: none">• 普通IO转超高IO• 超高IO转普通IO• 高IO转普通IO 以上情况仅为枚举, 存储类型均不支持变更。

8.6.4 数据超过了 RDS 实例的最大存储容量怎么办

问题症状

RDS数据库的磁盘空间满, 导致实例变为只读状态, 或者应用无法对RDS数据库进行读写操作, 从而影响业务。

问题原因

1. 数据空间占用过大。
2. 由于大量的事务和写入操作产生了大量的MySQL binlog日志文件。
3. 应用中存在大量的排序查询, 产生了过多的临时文件。

解决方案

1. 针对数据空间过大, 如果实例变为只读状态, 您可以先联系技术人员解除只读状态, 再删除无用的历史表数据进行释放空间(清理整张表使用DROP或TRUNCATE操作; 删除部分数据, 使用DELETE操作。如果是执行DELETE操作, 需要使用OPTIMIZE TABLE来释放空间); 如果实例非只读状态, 则可以直接执行删除操作; 如果没有可删除的历史数据, 需要进行磁盘扩容。
2. 如果是MySQL binlog日志文件占用过多, 可以联系技术人员清除本地的MySQL binlog日志, 来释放磁盘空间。
3. 针对大量排序查询导致的临时文件过大, 建议进行优化SQL查询。

4. 如果上述几种解决方案无效，就需要采用分库分表的方案。

8.7 安装客户端

8.7.1 如何安装 MySQL 客户端

MySQL官网提供了针对不同操作系统的客户端安装包，以5.6为例。单击[此处](#)下载MySQL最新版本，单击[此处](#)下载其他归档版本。下面将以Red Hat Linux为例介绍如何获取相应安装包及完成安装。

操作步骤

步骤1 获取安装包。

在下载页面找到对应版本[链接](#)，以MySQL-client-5.6.31-1.el6.x86_64.rpm为例，即可下载安装包。

图 8-2 下载

MySQL Product Archives

MySQL Community Server (Archived Versions)

Please note that these are old versions. New releases will have recent bug fixes and features!
To download the latest release of MySQL Community Server, please visit MySQL Downloads.

Product Version: 5.6.31
Operating System: Red Hat Enterprise Linux / Oracle Linux
OS Version: Red Hat Enterprise Linux 6 / Oracle Linux 6 (x86, 64-bit)

Package Name	Release Date	Size	Action
RPM Bundle (MySQL-5.6.31-1.el6.x86_64.rpm-bundle.tar)	May 19, 2016	215.4M	Download
RPM Package, MySQL Server (MySQL-server-5.6.31-1.el6.x86_64.rpm)	May 18, 2016	54.3M	Download
RPM Package, Client Utilities (MySQL-client-5.6.31-1.el6.x86_64.rpm)	May 18, 2016	17.9M	Download
RPM Package, Development Libraries (MySQL-devel-5.6.31-1.el6.x86_64.rpm)	May 18, 2016	3.2M	Download

步骤2 将安装包上传到弹性云服务器。

1. 创建弹性云服务器时，要选择操作系统，例如Redhat6.6，并为其绑定EIP（Elastic IP，弹性公网IP）。
2. 使用远程连接工具通过绑定的EIP连接到弹性云服务器，即可将安装包上传到弹性云服务器。

步骤3 执行以下命令安装MySQL客户端。

```
sudo rpm -ivh MySQL-client-5.6.31-1.el6.x86_64.rpm
```

说明

- 如果安装过程中报conflicts，可增加replacefiles参数重新安装，如下：
rpm -ivh --replacefiles MySQL-client-5.6.31-1.el6.x86_64.rpm
- 如果安装过程中提示需要安装依赖包，可增加nodeps参数重新安装，如下：
rpm -ivh --nodeps MySQL-client-5.6.31-1.el6.x86_64.rpm

----结束

8.7.2 如何安装 PostgreSQL 客户端

PostgreSQL官网提供了针对不同操作系统的[客户端安装方法](#)。

下面将以CentOS下PostgreSQL 12版本为例，介绍安装PostgreSQL客户端。

操作步骤

步骤1 登录弹性云服务器。

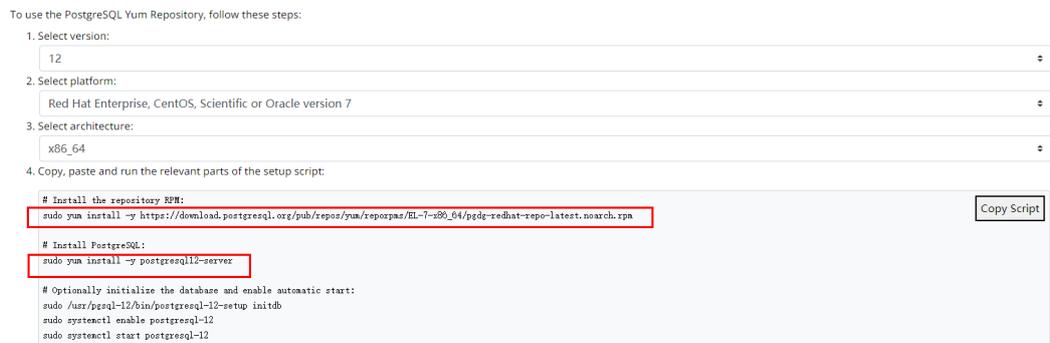
1. 创建弹性云服务器时，要选择操作系统，例如CentOS 7，并为其绑定EIP（Elastic IP，弹性公网IP）。
2. 使用远程连接工具通过绑定的EIP连接到弹性云服务器。

步骤2 打开**客户端安装**页面。

步骤3 选择数据库版本、操作系统、操作系统架构，在弹性云服务器上执行以下命令安装PostgreSQL客户端。

```
sudo yum install -y https://download.postgresql.org/pub/repos/yum/repopms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
sudo yum install -y postgresql12-server
```

图 8-3 安装客户端



- 数据库版本，选择与RDS for PostgreSQL数据库实例匹配的版本。
- 操作系统，选择与弹性云服务器匹配的操作系统。
- 操作系统架构，选择与弹性云服务器匹配的操作系统架构。

图 8-4 安装 rpm 包

```
root@ecs-d685 ~]# sudo yum install -y https://download.postgresql.org/pub/repos/yum/reposrpm/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
Loaded plugins: fastestmirror
pgdg-redhat-repo-latest.noarch.rpm                               | 8.6 kB  00:00:00
Examining /var/tmp/yum-root-ZonITG/pgdg-redhat-repo-latest.noarch.rpm: pgdg-redhat-repo-42.0-28.noarch
Marking /var/tmp/yum-root-ZonITG/pgdg-redhat-repo-latest.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package pgdg-redhat-repo.noarch 0:42.0-28 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch           Version         Repository              Size
=====
Installing:
pgdg-redhat-repo      noarch         42.0-28         /pgdg-redhat-repo-latest.noarch 13 k
=====
Transaction Summary
=====
Install 1 Package

Total size: 13 k
Installed size: 13 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Warning: RPMDB altered outside of yum.
Installing : pgdg-redhat-repo-42.0-28.noarch                1/1
Verifying  : pgdg-redhat-repo-42.0-28.noarch                1/1

Installed:
pgdg-redhat-repo.noarch 0:42.0-28

Complete!
```

图 8-5 客户端安装完成

```
Total                                                                 467 kB/s | 14 MB  00:00:30
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG
Importing GPG key 0x442DF0F8:
  Userid   : "PostgreSQL RPM Building Project <pgsql-pkg-yum@postgresql.org>"
  Fingerprint: 60c9 e2b9 1a37 d136 fe74 d176 1f16 d2e1 442d f0f8
  Package   : pgdg-redhat-repo-42.0-28.noarch (@/pgdg-redhat-repo-latest.noarch)
  From      : /etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : libicu-58.2-4.e17_7.x86_64                    1/4
  Installing : postgresql12-libs-12.13-1PGDG.rhel7.x86_64   2/4
  Installing : postgresql12-12.13-1PGDG.rhel7.x86_64        3/4
  Installing : postgresql12-server-12.13-1PGDG.rhel7.x86_64 4/4
  Verifying  : postgresql12-libs-12.13-1PGDG.rhel7.x86_64   1/4
  Verifying  : postgresql12-12.13-1PGDG.rhel7.x86_64        2/4
  Verifying  : postgresql12-server-12.13-1PGDG.rhel7.x86_64 3/4
  Verifying  : libicu-58.2-4.e17_7.x86_64                    4/4

Installed:
postgresql12-server.x86_64 0:12.13-1PGDG.rhel7

Dependency Installed:
libicu.x86_64 0:58.2-4.e17_7      postgresql12.x86_64 0:12.13-1PGDG.rhel7      postgresql12-libs.x86_64 0:12.13-1PGDG.rhel7

Complete!
```

步骤4 连接PostgreSQL实例。

图 8-6 连接成功

```
root@ecs-d685 ~]# psql -h [redacted] -d postgres -U root
Password for user root:
psql (12.13, server 12.11)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=>
```

----结束

8.7.3 如何安装 SQL Server Management Studio

Microsoft SQL Server官网提供了SQL Server Management Studio的安装包。SQL Server Management Studio应用程序只能在Windows环境运行。

操作步骤

步骤1 获取SQL Server Management Studio的安装包。

请访问[Microsoft网站](#)，以SQL Server Management Studio 18.0为例，下载安装包。

步骤2 将安装包上传到弹性云服务器。

步骤3 双击安装包，按照向导完成安装。

---结束

8.8 备份与恢复

8.8.1 RDS 能够保存多长时间的备份

关系型数据库实例的自动备份有效期根据用户设置的备份天数而定。手动备份没有时间限制，用户可根据需要进行删除。

备份存储在对象存储服务上，不占用您创建的数据库空间。

8.8.2 数据库在备份窗口中是否可用

备份窗口是用户定义的时间段，在该时间段内关系型数据库实例将进行备份。关系型数据库服务借助这些定期数据备份，让您能够将关系型数据库实例还原到保留期内的备份点。在备份时段期间，业务不受影响，但不能在关系型数据库服务的管理控制台做重启操作。

8.8.3 如何将 RDS 数据库备份到上

您可以通过导出SQL语句的方式将数据库备份到弹性云服务器上。弹性云服务器不限制存放哪些数据，但是数据必须符合国家法律法规。您可以在弹性云服务器上存放数据库备份，但不建议将弹性云服务器作为数据库备份空间使用。强烈推荐使用关系型数据库的备份功能将备份数据存放到专业的对象存储服务上，以获得更高的数据可靠性和服务保障。

8.8.4 为什么自动备份会失败

出现自动备份失败可能有以下原因：

1. 网络环境稳定性不足。如网络延时、中断，系统会识别并延时半小时再次执行备份，您也可以及时进行一次手动备份。
2. 系统多任务执行复杂性。如任务等待、中止，系统会识别并延时半小时再次执行备份，您也可以及时进行一次手动备份。
3. 实例状态异常。如实例故障、状态变更中，系统会识别并在状态恢复正常后再次执行备份，您也可以及时进行一次手动备份。
4. 数据库中表的数量会影响备份速度。表数量超过50万，备份将会失败。
5. 参数修改异常。如修改参数并关联实例后导致实例故障，您可以对比参数修改前后的值是否正确、修改的参数是否存在关联参数需要一并修改、或者尝试恢复默认参数重启实例。
6. 导入数据异常。

如导入数据不当导致系统表记录丢失：

7. 如果以上操作仍无法解决问题，请联系技术支持。

8.8.5 删除 RDS 实例后数据库备份会怎样

实例删除后，自动备份会被同步删除，手动备份不会被删除。

8.8.6 如果删除云账号备份是否会被删除

一旦删除云账号，自动备份和手动备份的数据也随之删除。

8.8.7 为什么数据表会丢失或数据被删除

RDS不会删除和操作用户的任何数据。出现这种情况，请检查是否为误操作，必要时可利用已有备份恢复文件。

检查误操作：如果已经，可通过审计日志查看数据执行记录。

已有备份恢复文件：

- 使用RDS实例的恢复功能。
- 将备份数据从弹性云服务器导入关系型数据库。

8.9 数据库监控

8.9.1 用户平时需要关注实例的哪些监控指标

用户需要关注的监控指标有：CPU利用率、内存利用率、磁盘空间利用率。

可以根据实际应用场景配置告警提示，当收到告警，可采取相应措施消除告警。

配置示例：

- 如果在某段时间内（如5min），CPU的利用率出现多次（如3次）不低于某特定值（如90%）的情况，则发出相应Cloud Eye告警。
- 如果在某段时间内（如5min），内存的利用率出现多次（如4次）不低于某特定值（如90%）的情况，则发出相应Cloud Eye告警。
- 如果在某段时间内（如5min），磁盘的使用率出现多次（如5次）不低于某特定值（如85%）的情况，则发出相应Cloud Eye告警。

📖 说明

Cloud Eye告警详细配置方法，请参见《云监控用户指南》中“创建告警规则”。

采取措施：

- 收到与CPU利用率、内存利用率有关的告警，通过实例规格变更分别增大CPU、内存。
- 收到与磁盘空间利用率有关的告警：
 - 调查磁盘空间消耗，查看是否可以从实例中删除数据或是将数据存档到其他系统以释放空间。
 - 通过磁盘空间扩容增大磁盘空间。

8.10 扩容及规格变更

8.10.1 RDS 实例在扩容期间是否一直可用

目前，关系型数据库服务支持磁盘扩容和CPU/内存规格变更。

- 在磁盘扩容期间，关系型数据库实例一直可用，业务不受影响，但不允许删除和重启实例。
- 在CPU/内存规格变更期间，有1到2次网络秒级闪断（RDS for SQL Server 2017企业版暂不支持闪断升级，请暂停业务进行规格变更）。如果是主备实例，规格变更过程中可能会发生主备倒换，倒换时业务可能会短暂中断。变更CPU/内存规格大约需要5~15分钟。

修改CPU/内存后，将会重启数据库实例。请选择业务低峰期，避免业务异常中断。重启后实例会自动释放内存中的缓存，请在业务低峰期进行重启，避免对高峰期业务造成影响。

8.10.2 为什么端口变更后实例状态异常

现象描述

- 端口变更后该实例状态为“故障”。
- 使用变更后端口号无法连接实例。

可能原因

提交的数据库端口被临时占用。

处理步骤

- 若变更成功，说明所需变更端口被临时占用，导致变更失败。
- 若仍然变更失败，请联系技术人员。

8.11 数据库参数修改

8.11.1 哪些参数设置不合理会导致数据库不可用（PostgreSQL）

由于参数配置不合理导致数据库不可用的情况有以下几种。

- 参数取值范围和实例规格有关：
参数shared_buffers、max_connections可配置的最大值和实例的实际物理内存大小有关，设置不合理，会导致数据库不可用。
- 参数之间存在关联关系：
 - 如果参数log_parser_stats、log_planner_stats或log_executor_stats存在配置为开启的情况，则需要将参数log_statement_stats配置为关闭，否则会导致数据库不可用。
 - 参数max_connections、autovacuum_max_workers和max_worker_processes配置需满足以下规则，否则会导致数据库不可用。

$\text{max_connections参数值} + \text{autovacuum_max_workers参数值} + \text{max_worker_processes参数值} + 1 < 8388607$

📖 说明

各参数的详细说明请参见[PostgreSQL官网](#)。

解决方法：

1. 登录控制台，通过查询近期日志，确定错误的参数。
2. 在参数配置页面，将参数值恢复成默认参数值，并重启数据库。
3. 将除错误参数以外的其他参数值设置为恢复之前的参数值，并将配置错误的参数调整为合理的参数值。

8.11.2 如何设置 SQL Server 添加 ndf 文件的路径

用户添加自定义数据库ndf辅助文件和tempdb数据库ndf辅助文件时，需要将ndf辅助文件分别放置在“D:\RDSDBDATA\DATA”路径和“D:\RDSDBDATA\Temp”路径下，不能放置在C盘，否则会导致系统盘空间被占满，服务可能会被停止。

8.11.3 RDS for MySQL 密码过期策略

MySQL 5.7和8.0版本支持通过设置全局变量“default_password_lifetime”来控制用户密码的默认过期时间。

参数“default_password_lifetime”的值为N，表示密码N天后过期，单位为天。默认值为0，表示创建的用户密码永不过期。

```
mysql> show variables like 'default_password_lifetime';
+-----+-----+
| Variable_name          | Value |
+-----+-----+
| default_password_lifetime | 0     |
+-----+-----+
1 row in set (0.00 sec)
```

修改全局自动密码过期策略

- 您可以在云数据库RDS界面，通过设置参数“default_password_lifetime”的值，修改密码过期策略。
- 通过命令修改全局变量“default_password_lifetime”的值。
`mysql> set global default_password_lifetime=0;`

查看当前所有用户的密码过期时间

执行以下命令：

```
mysql> select
user,host,password_expired,password_last_changed,password_lifetime from
user;
```

```
mysql> select user,host,password_expired,password_last_changed,password_lifetime from user;
+-----+-----+-----+-----+-----+
| user      | host      | password_expired | password_last_changed | password_lifetime |
+-----+-----+-----+-----+-----+
| mysql.session | localhost | N                | 2020-01-17 15:02:23   | NULL              |
| mysql.sys     | localhost | N                | 2020-01-17 15:02:23   | NULL              |
| rdsAdmin      | localhost | N                | 2020-01-17 15:02:30   | 0                 |
| root          | %         | N                | 2020-03-05 14:23:54   | NULL              |
| rdsRepl       | 192.168.% | N                | 2020-01-17 15:02:45   | 0                 |
| rdsMetric     | 192.168.% | N                | 2020-01-17 15:02:30   | 0                 |
| rdsBackup     | localhost | N                | 2020-01-17 15:02:30   | 0                 |
| u_test01      | %         | N                | 2020-03-05 14:28:10   | 30                |
| u_test02      | %         | N                | 2020-03-05 14:28:38   | NULL              |
| jeffrey       | localhost | N                | 2020-03-05 15:23:17   | NULL              |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

查看指定用户的密码过期策略

执行以下命令：

```
mysql> show create user jeffrey@'localhost';
```

```
mysql> show create user jeffrey@'localhost';
+-----+-----+-----+-----+-----+
| CREATE USER for jeffrey@localhost |
+-----+-----+-----+-----+-----+
| CREATE USER 'jeffrey'@'localhost' IDENTIFIED WITH 'mysql_native_password' AS '*1369F151658FC98255853119A9CBB0554DB807F' REQUIRE NONE PASSWORD EXPIRE DEFAULT ACCOUNT UNLOCK |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

“EXPIRE DEFAULT”表示遵从全局到期策略。

设置指定用户的密码过期策略

- 创建用户的同时设置密码过期策略
create user 'script'@'localhost' identified by '***' password expire interval 90 day;**
- 创建用户后设置密码过期策略
ALTER USER 'script'@'localhost' PASSWORD EXPIRE INTERVAL 90 DAY;
- 设置密码永不过期
mysql> CREATE USER 'mike'@'%' PASSWORD EXPIRE NEVER;
mysql> ALTER USER 'mike'@'%' PASSWORD EXPIRE NEVER;
- 设置密码遵从全局到期策略
mysql> CREATE USER 'mike'@'%' PASSWORD EXPIRE DEFAULT;
mysql> ALTER USER 'mike'@'%' PASSWORD EXPIRE DEFAULT;

8.12 日志管理

8.12.1 RDS for MySQL 慢日志延迟时间是多久

一般情况下是5分钟，如果五分钟内慢日志达到10M会提前上传。

8.12.2 RDS for SQL Server 慢查询需要多长时间

RDS for SQL Server慢查询时间指标为5秒。

8.12.3 如何通过命令获取 Microsoft SQL Server 的错误日志

步骤1 通过rdsuser账户登录Microsoft SQL Server客户端。

步骤2 在查询框进行执行如下查询语句，查询错误日志。

```
EXECUTE master.dbo.rds_read_errorlog  
FileID,LogType,FilterText,FilterBeginTime,FilterEndTime
```

- FileID: Error Log的日志文件编号，0表示最近的日志。
- LogType: 日志类型。取值为1查询的日志类型为error logs，取值为2查询的日志类型为Agent logs。
- FilterText: 查询关键字，可为NULL。
- FilterBeginTime: 日志查询开始时间，可为NULL。
- FilterEndTime: 日志查询完成时间，可为NULL。

示例如下：

```
EXEC master.dbo.rds_read_errorlog 0,1,'FZYUN','2018-06-14 14:30','2018-06-14  
14:31'
```

查询结果，如图8-7所示：

图 8-7 查询示例

	LogDate	ProcessInfo	Text
1	2018-06-14 14:30:47.490	spid64	Starting up database 'FZYUN032020'.
2	2018-06-14 14:30:47.430	spid64	CHECKDB for database 'FZYUN029029' finished wit...
3	2018-06-14 14:30:47.400	spid64	Starting up database 'FZYUN029029'.
4	2018-06-14 14:30:47.330	spid64	CHECKDB for database 'FZYUN029027' finished wit...
5	2018-06-14 14:30:47.290	spid64	Starting up database 'FZYUN029027'.
6	2018-06-14 14:30:47.220	spid64	CHECKDB for database 'FZYUN02' finished without...
7	2018-06-14 14:30:47.180	spid64	Starting up database 'FZYUN02'.
8	2018-06-14 14:30:47.110	spid64	CHECKDB for database 'FZYUN' finished without e...
9	2018-06-14 14:30:47.080	spid64	Starting up database 'FZYUN'.
10	2018-06-14 14:30:46.840	spid64	Starting up database 'FZYUN032020'.

----结束

8.12.4 RDS 慢日志统计分析是否可以导出

RDS慢日志统计分析不支持导出。

8.13 网络安全

8.13.1 RDS 有哪些安全保障措施

网络

- 关系型数据库实例可以设置所属虚拟私有云，从而确保关系型数据库实例与其他业务实现网络安全隔离。
- 使用安全组确保访问源为可信的。

- 使用SSL通道，确保数据传输加密。

管理

通过统一身份认证服务（Identity and Access Management，简称IAM），可以实现对关系型数据库实例的管理权限控制。

8.13.2 如何确保在 VPC 中运行的 RDS 实例的安全

虚拟私有云的安全组可用来帮助确保虚拟私有云内关系型数据库实例的安全。此外，通过网络访问控制列表（ACL），可以允许或拒绝进入和退出各个子网的网络流量。

8.13.3 如何保障 EIP 数据传输安全

使用EIP连接数据库时，业务数据会在公网上进行传输，数据容易泄露，因此强烈建议您使用关系型数据库服务的SSL通道来对公网上传输的数据进行加密，防止数据泄露。您也可以借助云专线或虚拟专用网络来完成数据传输通道的加密。

8.13.4 如何防止任意源连接数据库

- 数据库开放EIP后，如果公网上的恶意人员获取到您的EIP DNS和数据库端口，那么便可尝试破解您的数据库并进行进一步破坏。因此，强烈建议您保护好EIP DNS、数据库端口、数据库账号和密码等信息，并通过关系型数据库实例的安全组限定源IP，保障只允许可信源连接数据库。
- 为避免恶意人员轻易破解您的数据库密码，请按照关系型数据库实例的密码策略设置足够复杂度密码，并定期修改。
- 对于支持防暴力破解的RDS for SQL Server数据库来说，恶意人员一旦获取到您的EIP DNS、数据库端口、数据库账号信息，而尝试暴力破解，那么可能导致您的正常业务连接被删除。此情况下，您可通过限制连接源并修改数据库账号和密码来避免进一步攻击破坏。

说明

RDS for MySQL、RDS for PostgreSQL不支持防暴力破解。

RDS for SQL Server默认开启防暴力破解，不提供关闭防暴力破解的功能。

8.13.5 将根证书导入 Windows/Linux 操作系统

导入 Windows 操作系统

1. 单击“开始”，运行框输入“MMC”，回车。
2. 在MMC控制台菜单栏中单击“文件”，选择“添加/删除管理单元”。
3. 在“添加或删除管理单元”对话框，选择“可用管理单元”区域的“证书”。单击“添加”添加证书。
4. 在“证书管理”对话框，选择“计算机账户”，单击“下一步”。
5. 在“选择计算机”对话框，单击“完成”。
6. 在“添加或删除管理单元”对话框，单击“确定”。
7. 在MMC控制台，双击“证书”。
8. 右键单击“受信任的根证书颁发机构”，选择“所有任务”，单击“导入”。
9. 单击“下一步”。

10. 单击“浏览”，将文件类型更改为“所有文件 (*.*)”。
11. 找到下载的根证书ca.pem文件，单击“打开”，然后在向导中单击“下一步”。

须知

您必须在浏览窗口中将文件类型更改为“所有文件 (*.*)”才能执行此操作，因为“.pem”不是标准证书扩展名。

12. 单击“下一步”。
13. 单击“完成”。
14. 单击“确定”，完成根证书导入。

导入 Linux 操作系统

您可以使用任何终端连接工具（如WinSCP、PuTTY等工具）将证书上传至Linux系统任一目录下。

8.13.6 如何确认 SSL 根证书的有效期

如果您正在使用SSL连接MySQL，执行如下命令查看证书有效期起止时间，判断证书是否到期。

```
show status like '%ssl_server%';
```

在证书到期前请及时更新根证书为最新版本：

1. 在RDS“实例管理”页面，单击实例名称进入“基本信息”页面，单击“数据库信息”模块“SSL”处的，下载新的根证书文件或根证书捆绑包。
2. 重启实例以使新的根证书生效。
3. 使用新的根证书文件或根证书捆绑包连接实例。

说明

用户证书即将到期时要及时更换正规机构颁发的证书，提高系统安全性。

8.13.7 如何甄别数据库中数据被损坏

- 数据被篡改

有很多安全措施保证只有经过认证的用户才能操作数据库表数据，只能通过指定的数据库服务端口来访问数据库表。

主备数据库在同步过程中有传输包校验功能，所以不会出现数据被篡改。MySQL使用InnoDB引擎，不容易出现数据损坏。

- 可能会出现实例服务器突然停电，导致数据库有页损坏，数据库无法启动。

当主机出现故障时，系统会在1~5分钟切换到备机，继续为您服务。切换过程中关系型数据库实例不可访问，需要您设置好程序跟服务的自动重连，避免因为切换导致服务不可用。

9 故障排除

9.1 RDS for MySQL

9.1.1 备份恢复

9.1.1.1 备份期间 DDL 操作导致备份失败

场景描述

实例连续两天备份失败，备份时间窗内有DDL操作。

问题原因

MySQL全量备份基于xtrabackup，为保证数据一致性，全量备份操作与DDL操作存在元数据锁冲突，会导致备份一直阻塞等待，超时失败。

执行“show processlist”命令，排查业务侧在备份时间窗内是否有DDL操作。

解决方案

1. 停止相应的DDL操作后，重试手动备份。
2. 建议此后的DDL业务变更操作应尽量避免备份窗口。

9.1.1.2 RDS for MySQL 备份任务失败分析思路

场景描述

客户使用RDS for MySQL数据库服务，使用自有备份脚本，通过mysqldump命令进行异机备份。备份数据到一台与RDS在不同子网的ECS主机，但备份任务运行300秒后会中断，无法完成备份任务。

原因分析

为了排除网络问题，将执行备份任务的ECS主机更换为一台与RDS服务器同一子网的ECS主机上，备份任务执行成功。

- 网络排查反馈：跨子网只是在虚拟网络设备中转换，类似于网关，时延和带宽没有大的差异。
- 数据库排查反馈：每次中断都是300秒就中断了，是因为RDS服务器端的参数导致，如果客户端300秒内没有完成数据写入，就会根据参数“net_write_timeout”的值断开数据库连接。

```
| net_write_timeout | 300 |
```

处理过程

步骤1 了解mysqldump的备份的数据流向，采用的协议和端口。

mysqldump使用TCP协议连接RDS服务器的8635端口，建立连接后，通过网络将数据备份到本地磁盘。

步骤2 分析两台ECS主机的差异点：

1. 查看两台主机硬件配置是否一致：同为2CORE 6GB。
2. 查看两台主机OS版本是否一致：同为centos7.4。

步骤3 查看网卡速率是否一致。

步骤4 查看内核参数配置是否一致，发现备份失败主机未做优化，备份成功的主机有做网络参数优化。

```
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
vm.swappiness = 8
net.ipv4.neigh.default.gc_stale_time=120
net.ipv4.conf.all.rp_filter=0
net.ipv4.conf.default.rp_filter=0
net.ipv4.conf.default.arp_announce = 2
net.ipv4.conf.lo.arp_announce=2
net.ipv4.conf.all.arp_announce=2
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_synack_retries = 2
vm.swappiness=5
net.ipv4.tcp_fack=1
net.ipv4.tcp_tw_reuse=1
net.ipv4.tcp_tw_recycle=1
net.ipv4.tcp_fin_timeout=30
net.ipv4.tcp_keepalive_time=600
net.ipv4.tcp_keepalive_probes=5
net.ipv4.tcp_keepalive_intvl=15
net.ipv4.tcp_max_syn_backlog=4096
net.ipv4.tcp_rmem = 4096 65536 65536
net.core.rmem_default=434176
net.core.wmem_default=434176
net.core.rmem_max=1048576
net.core.wmem_max=1048576
kernel.shmmax=5153960755
kernel.sem= 512 524288 32 1024
kernel.msgmni=128
net.core.somaxconn=20000
fs.file-max=800000
net.ipv4.tcp_mcm = 524288 706432 1048576
net.ipv4.tcp_wmem = 4096 16384 65536
net.ipv4.tcp_rmem = 4096 83700 65536
```

步骤5 初步判断问题与TCP缓存参数设置相关，将成功ECS主机的内核参数覆盖到问题ECS主机上并使其生效，运行备份任务，备份任务未再中断，成功完成了备份。

```
[root@szst-test01 data]# tail -fn 20 database_dump.20180413.sql
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2018-04-13 13:41:40
```

----结束

解决方案

mysqldump在备份过程中会产生大数据读写，本场景是跨主机通过网络异机备份，备份端数据写入能力和TCP缓存无法匹配RDS端的发送能力，超时时间达到数据库写超时设定的300秒，最终导致备份中断。可以通过修改内核参数增加TCP的缓存，提高备份端网络处理能力来解决问题。

9.1.1.3 手动下发全量备份比自动下发全量备份时间长

场景描述

周期性触发的自动全备比客户手动触发的全备耗时短，且备份数据量接近，均为20GB。

原因分析

查看磁盘吞吐，发现手动备份的磁盘吞吐低于自动备份的磁盘吞吐。

查看CES指标硬盘读吞吐量、硬盘写吞吐量。

备份任务会依赖OBS服务，自动备份处于OBS的业务低峰期，所以耗时短，而手动备份恰好处于OBS业务的高峰期，所以耗时长。

解决方案

建议客户避开OBS高峰期进行全量备份，可以缩短备份时间。

9.1.1.4 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RDS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。

9.1.1.5 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，也可能会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 -h 192.168.0.199 --set-gtid-purged=OFF --single-transaction --flush-logs -q test t1>t1.sql
```

9.1.1.6 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 配置项说明

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。

选项名称	说明
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 -h 192.168.0.199 --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 -h 192.168.0.199 --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 -h 192.168.0.199 --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 -h 192.168.0.199 --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 -h 192.168.0.199 -P8635 --set-gtid-purged=OFF -F -n -t -d -E -R db1 > others.sql
```

9.1.1.7 怎么解决执行mysqldump出现SET @@SESSION.SQL_LOG_BIN等SQL的问题

场景描述

执行mysqldump时，会出现如下如所示代码。

图 9-1 代码显示

```
1  -- MySQL dump 10.13 Distrib 5.7.24, for Linux (x86_64)
2  --
3  -- Host: 192.168.1.64 Database: rptdb
4  -----
5  -- Server version  5.7.31-2-log
6
7  /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
8  /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
9  /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
10 /*!40101 SET NAMES utf8 */;
11 /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
12 /*!40103 SET TIME_ZONE='+00:00' */;
13 /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
14 /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
17 SET @MYSQLDUMP_TEMP_LOG_BIN = @@SESSION.SQL_LOG_BIN;
18 SET @@SESSION.SQL_LOG_BIN= 0;
19
20
21 --
22 -- GTID state at the beginning of the backup
23 --
24 SET @@GLOBAL.GTID_PURGED='7f47edf7-2e4d-11eb-9a43-fa163eac6f0:1-36975382';
25
26 --
27 -- Dumping routines for database 'rptdb'
28 --
29 /*!50003 DROP FUNCTION IF EXISTS `f_sys_get_partition` */;
30 /*!50003 SET @saved_cs_client          = @@character_set_client */;
31 /*!50003 SET @saved_cs_results        = @@character_set_results */;
32 /*!50003 SET @saved_col_connection    = @@collation_connection */;
33 /*!50003 SET character_set_client     = utf8 */;
34 /*!50003 SET character_set_results     = utf8 */;
35 /*!50003 SET collation_connection     = utf8_general_ci */;
36 /*!50003 SET @saved_sql_mode         = @@sql_mode */;
```

故障分析

开启了“gtid-mode=ON”参数。

如果一个数据库开启了GTID，使用mysqldump备份或者转储的时候，即使不是RDS for MySQL全库(所有库)备份，也会备份整个数据库所有的GTID号。

解决方案

RDS for MySQL数据库在主从数据库进行导出备份和恢复的时候，需要注意是否启用数据库用GTID模式。

如果开启，则在mysqldump数据时，应该在mysqldump命令加上参数“-set-gtid-purged=OFF”。

9.1.1.8 mysqldump 导出数据报错权限不足

场景描述

mysqldump使用指定用户导出数据库数据时，报错：'Access denied; you need (at least one of) the PROCESS privilege(s)

```
mysql> mysqldump -h 192.168.1.64 -P 3306 -u zzkj -p zzkj --set-gtid-purged=off --skip-lock-tables zzkj > zzkj.sql
mysqldump: [warning] Using a password on the command line interface can be insecure.
mysqldump: Error: 'Access denied; you need (at least one of) the PROCESS privilege(s) for this operation' when trying to dump tablespaces
```

原因分析

mysqldump使用指定用户导出数据时，需要赋予PROCESS权限。

解决方案

使用管理员账户给相应用户授予PROCESS权限。

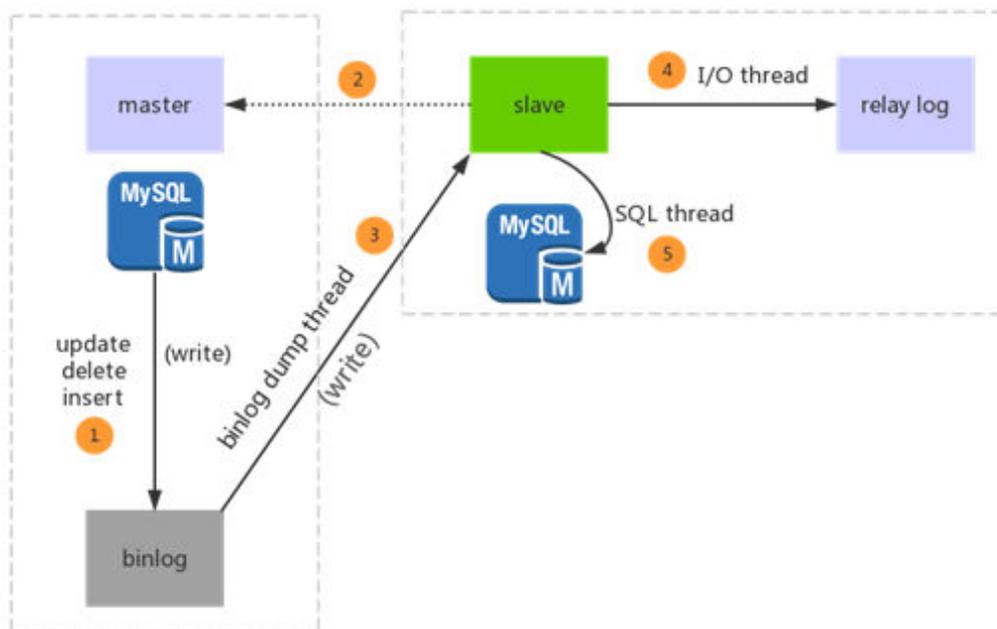
```
GRANT SELECT, PROCESS ON *.* TO 'dump_user' @' %' ;  
FLUSH PRIVILEGES;
```

9.1.2 主备复制

9.1.2.1 MySQL 主备复制原理简介

RDS for MySQL的默认备库、只读实例、自建从库、DRS链路灾备实例均采用MySQL的Binlog复制技术，也称为MySQL主备复制或主从复制技术。本章节介绍MySQL的主从复制原理。

主备复制流程



- 主节点（Master）中有数据更新时，会按照Binlog格式，将更新的操作以event形式写入到主节点的Binlog中。event有多种类型：INSERT、DELETE、UPDATE、QUERY等。
- 从节点（Slave）连接主节点时，有多少个从节点就会创建多少个Binlog dump线程。
- 当主节点的Binlog发生变化时，Binlog dump线程会通知所有从节点，并将相应Binlog内容推给从节点。
- 从节点的I/O thread收到Binlog内容后，会将内容写到本地relay log（中继日志）。
- 从节点的SQL thread会读取I/O thread写入的relay log，并且根据relay log中的event，回放对应的操作（DML、DDL等）。

Seconds_Behind_Master 计算方式

Seconds_Behind_Master即主备复制时延，通过show slave status查询获取。Seconds_Behind_Master计算的伪代码实现如下：

```
if (SQL thread is running)
//如果SQL线程启动
{
  if (SQL thread processed all the available relay log)
  //IO thread拉取主库Binlog的位置和sql thread应用的relay log相对于主库Binlog的位置相等
  {
    if (IO thread is running)
      //如果IO线程启动，设置延迟为0
      print 0;
    else
      //如果IO线程未启动，设置延迟为null
      print NULL;
  }
  else
    //如果SQL线程没有应用完所有的IO线程写入的event，那么需要计算Seconds_Behind_Master
    按公式计算Seconds_Behind_Master的值;
}
else
  //如果SQL线程也没有启动，则设置为空值
  print NULL;
```

上述伪代码中，Seconds_Behind_Master的计算公式为：

$$\text{Seconds_Behind_Master} = \text{time}(0) - \text{last_master_timestamp} - \text{clock_diff_with_master}$$

相关变量含义如下：

1. time(0)：当前从节点服务器的系统时间。
2. clock_diff_with_master：从节点的系统时间和主节点服务器系统时间的差值，一般为0。如果主从节点系统时间不一致，那么计算出的从节点复制时延会不准确。
3. last_master_timestamp：从节点在回放relay log中event过程中计算和更新，该变量在并行复制（MTS）和非并行复制方式下，更新的时机是不同的，默认全部开启并行复制：
 - 并行复制：可以简单理解为，从节点的SQL线程在每个事务执行完成后，更新last_master_timestamp值，其更新是以事务为单位。所以大事务、DDL容易导致主备延迟大，具体请参见[主备复制延迟持续增长后自动恢复](#)。
 - 非并行复制：从节点的SQL线程读取了relay log中的事务后，事务未执行前便会更新last_master_timestamp，其更新是以事务为单位。

综上所述，Seconds_Behind_Master的计算公式可以理解为：

$$\text{Seconds_Behind_Master} = \text{当前从节点服务器的系统时间} - \text{从节点SQL线程处理中事务在主节点的执行时间} - \text{从节点的系统时间和主节点服务器系统时间的差值}$$

9.1.2.2 主备复制延迟持续增长后自动恢复

场景描述

实例只读复制延迟很大，且在一段时间内持续增长，然后自动恢复。

查看CES指标实时复制时延，下图仅供参考：

原因分析

根据[MySQL主备复制延迟场景及解决方案](#)和[MySQL主备复制原理简介](#)的相关分析，可以推测此场景为大事务或DDL操作导致。

可以通过分析全量日志或慢日志，观察是否有大事务或DDL操作：

慢日志中有如下加索引的DDL操作，表的数据量上亿，耗时近一天，从而导致只读或备机在回放该DDL时复制延迟持续增长，回放完DDL后延迟恢复正常。

```
# Time: 2022-04-18 10:00:00.734954+08:00
# User@Host: root@ [192.168.0.185] Id: 20167391
# Query_time: 82714.322375 Lock_time: 0.000173 Rows_sent: 1 Rows_examined: 999999999 Thread_id: 20167391 Schema:
# Read_last: 0 Read_key: 0 Read_next: 0 Read_prev: 0 Read_rnd: 0 Read_rnd_next: 493972 Sort_merge_passes: 0
# Tables: 0 Created_tmp_tables: 0 Start: 2022-04-18 10:00:00.734954+08:00 End: 2022-04-18 10:00:00.734954+08:00
# QC_Hit: No Full_scan: No Full_join: No Tmp_table: No Tmp_table_on_disk: No Filesort: No Filesort_on_disk: No
SET timestamp=1649777004;
alter table user_information add index ind_name_no(name,no);
```

解决方案

- 该场景属于正常现象，等待DDL执行完成后，延迟会自动恢复。
- 建议在业务低峰期进行加索引的操作。

9.1.2.3 MySQL 主备复制延迟场景及解决方案

RDS for MySQL的默认备库、只读实例、自建从库、DRS链路灾备实例均基于MySQL的Binlog复制技术，也称为MySQL主备复制或主从复制技术。主备复制实现又分为异步复制或半同步复制，无论哪种方式，由于业务执行的语句的原因，不可避免的存在主备复制延迟。

现象表现为：RDS for MySQL备机或只读存在复制延迟过高，甚至产生复制延迟大的告警。

场景 1：主库执行了大事务

大事务一般指一个事务中包含大量的数据更新操作，例如一个事务包含几万次DML（insert, update, delete）操作、一条SQL语句批量更新了上万行数据等，大事务往往本身的执行时间很长（分钟级）。当主实例执行了大事务后，会产生大量的Binlog日志，备机或只读节点拉取这些Binlog耗时比一般事务长，且至少需要花费与主实例相同的时间来回放这些事务的更新，从而导致备机或只读节点出现复制延迟。

排查方法：

- 对于包含大量DML语句的大事务，使用如下命令，找到长时间执行的事务。
select t.*,to_seconds(now())-to_seconds(t.trx_started) idle_time from INFORMATION_SCHEMA.INNODB_TRX t \G;
- 对于一条SQL语句执行大量数据的大事务，执行**show full processlist**，查找是否存在长时间执行的delete或update语句。
- 分析全量日志或慢日志，检查是否有大事务。

解决方法：

- 为了保证主从数据的一致性，需要等待大事务执行完成，主备复制延迟才能恢复。
- 业务侧避免此类大事务，可以将大事务拆分为小事务，分批执行。例如，通过where条件或limit语句限制每次要更新的数据量。

场景 2：对无主键表更新

RDS for MySQL的Binlog采用row格式，对每一行的数据更新，都会形成row格式Binlog event记录。例如：一个update语句更新100行数据，那么row格式的Binlog中会形成100行update记录，备机或只读回放时会执行100次单行update。

只读节点和备机在回放主库的Binlog event时，会根据表的主键或者二级索引来检索需要更改的行。如果对应表未创建主键，则会产生大量的全表扫描，从而降低了Binlog日志的应用速度，产生复制延迟。

排查方法：

通过**show create table xxx**，分析执行慢的update和delete语句对应的表，分析是否有主键。

解决方法：

给无主键表增加主键，给缺少二级索引的表增加索引。

场景 3：DDL 操作

DDL操作往往执行时间很长，尤其是表数据量很大时。通常情况下，只读节点或备机回放一个DDL操作的时间和主库花费的时间基本一致。因此，当主机执行了大表的DDL操作后，备机和只读节点在回放该DDL期间，复制时间必然是一致增大的。

解决方法：

该场景为正常现象，等DDL执行完成后，主备复制延迟才能恢复。建议在业务低峰期执行DDL操作。

场景 4：只读实例等待 MDL 锁

只读实例上往往有业务流量，如果存在只读长事务正在执行，会阻塞主实例同步过来的相应表的DDL操作，卡在了表MDL锁获取，进而阻塞所有同表的后续Binlog回放，导致复制延迟越来越大。

排查方法：

1. 登录只读节点，使用如下命令，观察是否有长时间执行的事务。

```
select t.*,to_seconds(now())-to_seconds(t.trx_started) idle_time from INFORMATION_SCHEMA.INNODB_TRX t \G;
```
2. 查看只读节点的MDL锁视图，观察是否有MDL锁冲突。

```
select * from information_schema.metadata_lock_info;
```

根据MDL锁视图中的线程ID，找到阻塞的session。

解决方法：

kill只读节点上阻塞DDL操作的长事务，或者在业务侧提交该长事务。

场景 5：只读实例规格小于主实例

只读实例、DRS灾备实例的规格小于主实例时，一旦主实例写负载升高到一定程度，只读实例或DRS灾备实例会因为自身资源不足，无法及时回放Binlog，导致复制延迟增加。

解决方法：

只读实例或DRS灾备实例扩大规格，与主实例规格匹配。

9.1.2.4 RDS 主备复制关系异常

操作场景

有时候客户会遇到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** 在“实例管理”页面，选择指定的实例，单击实例名称。
- 步骤2** 在“基本信息”页面，单击目标安全组名称，进入实例安全组页面。
- 步骤3** 在“入方向规则”页签，单击“添加规则”，选择Any协议，源地址为自身安全组，即安全组的源端要有安全组自己。
- 步骤4** 添加完策略之后，主备复制的关系便会恢复正常。

----结束

9.1.2.5 主备复制时延瞬间飙升回落

场景描述

只读实例的CES监控上，出现只读复制时延会在某个瞬间升高并回落。

原因分析

- 此类问题与复制时延（Seconds_Behind_Master）的计算方式相关，关于复制时延的计算方式，详见[MySQL主备复制原理简介](#)。
- 出现复制时延尖峰是因为：只读节点IO线程刚好接收到了一个新的Binlog文件，而其SQL线程还没开始回放新的Binlog。导致计算复制时延的last_master_timestamp值还停留在上一个Binlog事务在主机的执行时间，与当前只读节点系统时间time(0)存在时间差，从而出现复制时延尖峰。当SQL线程开始解析新的Binlog时，复制时延立刻回落。
- 此类问题为偶现现象，不影响实际业务。

下载复制时延飙升回落时间段的Binlog，会发现如下现象：

新的Binlog的第一个事务执行时间与上一个Binlog最后一个事务的结束时间刚好与突增回落的时间差匹配。

```
at 63766
21092 16:06:03 server id 3559516408 end_log_pos 65793 CRC32 0x8cccd7df Rotate to mysql-bin.011018 pos: 4
SET @@SESSION.GTID_NEXT= 'AUTOMATIC' /* added by mysqlbinlog */ /*!*/;
DELIMITER ;
# End of log file
/*!50003 SET COMPLETION_TYPE=@OLD_COMPLETION_TYPE*/;
/*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=0*/;

~/Desktop/新建文件夹
mysqlbinlog --no-defaults -vv --base64-output=decode-rows mysql-bin.011018
/*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=1*/;
/*!50003 SET ROLD_COMPLETION_TYPE=@NEWCOMPLETION_TYPE,COMPLETION_TYPE=0*/;
DELIMITER /*!*/;
at 4
210923 16:06:03 server id 3559516408 end_log_pos 126 CRC32 0xd7e526a7 Start: binlog v 4, server v 5.7.32-2-log created 210923 16:06:03
at 126
210923 16:06:03 server id 3559516408 end_log_pos 197 CRC32 0x722a15a4 Previous-GTIDs
353f4e30-e856-11eb-b9c4-fa163ef03ae7:1-4640966
at 197
21092 16:09:16 server id 3559516408 end_log_pos 262 CRC32 0x6ec61958 GTID last_committed=0 sequence_number=1 rbr_only=yes
/*!50728 SET TRANSACTION ISOLATION LEVEL READ COMMITTED'/*!*/;
SET @@SESSION.GTID_NEXT= '353f4e30-e856-11eb-b9c4-fa163ef03ae7:4640967'/*!*/;
```

```
# at 65746
#210923 16:11:03 server id 3559516408 end_log_pos 65793 CRC32 0xe1f2a4fb Rotate to mysql-bin.011019 pos: 4
SET @@SESSION.GTID_NEXT= 'AUTOMATIC' /* added by mysqlbinlog */ /*!*/;
DELIMITER ;
# End of log file
/*!50003 SET COMPLETION_TYPE=@OLD_COMPLETION_TYPE*/;
/*!50030 SET @@SESSION.PSEUDO_SLAVE_MODE=0*/;

~/Desktop/新建文件夹
$ mysqlbinlog --no-defaults -vv --base64-output=decode-rows mysql-bin.011019
/*!50030 SET @@SESSION.PSEUDO_SLAVE_MODE=1*/;
/*!50003 SET @@OLD_COMPLETION_TYPE=@COMPLETION_TYPE,COMPLETION_TYPE=0*/;
DELIMITER /*!*/;
# at 4
#210923 16:11:03 server id 3559516408 end_log_pos 126 CRC32 0x6bcc3575 Start: binlog v 4, server v 5.7.32-2-log created 210923 16:11:03
# at 126
#210923 16:11:03 server id 3559516408 end_log_pos 197 CRC32 0xc8cb1f82 Previous-GTIDs
# 353f4e30-e856-11eb-b9c4-fa163ef03ae7:1-4640967
# at 197
#210923 16:14:16 server id 3559516408 end_log_pos 262 CRC32 0xcb9ab23b GTID last_committed=0 sequence_number=1 rbr_only=yes
/*!50718 SET TRANSACTION ISOLATION LEVEL READ COMMITTED*//*!*/;
SET @@SESSION.GTID_NEXT= '353f4e30-e856-11eb-b9c4-fa163ef03ae7:4640968'/*!*/;
```

解决方案

无需解决。该场景为RDS for MySQL主备复制机制中的正常行为，为偶现现象。

9.1.3 参数类

9.1.3.1 控制台上修改 long_query_time 参数后未生效

场景描述

在控制台将“long_query_time”参数值从0.1s修改为0.2s后，慢日志中还存在小于0.2s的慢SQL执行记录。

原因分析

控制台上修改“long_query_time”参数是全局级别生效，修改完后，后续新建连接会使用最新设置的参数，但是旧连接的“long_query_time”属性值不会被改变，仍然保持旧的值（该案例中是0.1s），所以小于0.2s的慢SQL是在旧连接上产生的。

出现该现象的原因是MySQL机制导致，所以不仅“long_query_time”参数会出现此类问题，其他控制台上可以修改的全局参数，也会发生类似现象：只有新建连接生效，旧连接不生效。

解决方案

如果想让某些会话连接采用最新的“long_query_time”值，关闭相应会话连接，重新建立连接即可生效。

9.1.3.2 GROUP_CONCAT 结果不符合预期

场景描述

SQL语句中使用GROUP_CONCAT()函数时，出现结果不符合预期的情况。

原因分析

GROUP_CONCAT()函数返回一个字符串结果，该结果由分组中的值连接组合而成。需要注意的是：这个函数的结果长度是有限制的，由group_concat_max_len参数决定。

示例：

```
mysql> show variables like 'group_concat_max_len';
+-----+-----+
| Variable_name      | Value |
+-----+-----+
| group_concat_max_len | 1024  |
+-----+-----+
1 row in set (0.01 sec)

mysql> select GROUP_CONCAT(c1,c2,c3) from dis;
+-----+
| GROUP_CONCAT(c1,c2,c3) |
+-----+
| 111,222,322           |
+-----+
```

```
mysql> set session group_concat_max_len=8;
Query OK, 0 rows affected (0.00 sec)

mysql> show variables like 'group_concat_max_len';
+-----+-----+
| Variable_name      | Value |
+-----+-----+
| group_concat_max_len | 8     |
+-----+-----+
1 row in set (0.01 sec)

mysql> select GROUP_CONCAT(c1,c2,c3) from dis;
+-----+
| GROUP_CONCAT(c1,c2,c3) |
+-----+
| 111,222,                |
+-----+
```

解决方案

调整group_concat_max_len参数值，适配GROUP_CONCAT()函数的结果长度。

9.1.3.3 RDS for MySQL 创建索引失败报错[ERROR] 1071 的解决方案

场景描述

创建索引长度超限制导致创建失败。创建表索引时出现如下错误：

[ERROR] 1071 - Specified key was too long; max key length is 3072 bytes

问题可能出现的版本：MySQL-8.0.20.5

故障分析

InnoDB表引擎有限制。

默认情况下，索引前缀长度限制为 767 字节，当开启了“innodb_large_prefix”选项时，索引前缀长度扩展到 3072 字节。

```
SHOW VARIABLES LIKE '%innodb_large_prefix%';
```

```
mysql> show variables like '%innodb_large_prefix%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_large_prefix | ON |
+-----+-----+
1 row in set (0.02 sec)
```

索引前缀长度还和 InnoDB 的 page size 有关。“innodb_page_size”选项默认是 16KB 的时候，最长索引前缀长度是 3072 字节，如果是 8KB 的时候，最长索引前缀长度是 1536 字节；4KB 的时候，最长索引前缀长度是 768 字节。

```
SHOW VARIABLES LIKE '%innodb_page_size%';
```

```
mysql> show variables like 'innodb_page_size';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| innodb_page_size | 16384 |
+-----+-----+
1 row in set (0.01 sec)
```

查看问题表结构，并查询所有支持的字符集及其字节占用情况：

```
SHOW CHARACTER SET;
```

```
mysql> SHOW CHARACTER SET;
```

Charset	Description	Default collation	Maxlen
armscii8	ARMSCII-8 Armenian	armscii8_general_ci	1
ascii	US ASCII	ascii_general_ci	1
big5	Big5 Traditional Chinese	big5_chinese_ci	2
binary	Binary pseudo charset	binary	1
cp1250	Windows Central European	cp1250_general_ci	1
cp1251	Windows Cyrillic	cp1251_general_ci	1
cp1256	Windows Arabic	cp1256_general_ci	1
cp1257	Windows Baltic	cp1257_general_ci	1
cp850	DOS West European	cp850_general_ci	1
cp852	DOS Central European	cp852_general_ci	1
cp866	DOS Russian	cp866_general_ci	1
cp932	SJIS for Windows Japanese	cp932_japanese_ci	2
dec8	DEC West European	dec8_swedish_ci	1
eucjpm	UJIS for Windows Japanese	eucjpm_japanese_ci	3
euckr	EUC-KR Korean	euckr_korean_ci	2
gb18030	China National Standard GB18030	gb18030_chinese_ci	4
gb2312	GB2312 Simplified Chinese	gb2312_chinese_ci	2
gbk	GBK Simplified Chinese	gbk_chinese_ci	2
geostd8	GEOSTD8 Georgian	geostd8_general_ci	1
greek	ISO 8859-7 Greek	greek_general_ci	1
hebrew	ISO 8859-8 Hebrew	hebrew_general_ci	1
hp8	HP West European	hp8_english_ci	1
keybcs2	DOS Kamenicky Czech-Slovak	keybcs2_general_ci	1
koi8r	KOI8-R Relcom Russian	koi8r_general_ci	1
koi8u	KOI8-U Ukrainian	koi8u_general_ci	1
latin1	cp1252 West European	latin1_swedish_ci	1
latin2	ISO 8859-2 Central European	latin2_general_ci	1
latin5	ISO 8859-9 Turkish	latin5_turkish_ci	1
latin7	ISO 8859-13 Baltic	latin7_general_ci	1
macce	Mac Central European	macce_general_ci	1
macroman	Mac West European	macroman_general_ci	1
sjis	Shift-JIS Japanese	sjis_japanese_ci	2
swe7	7bit Swedish	swe7_swedish_ci	1
tis620	TIS620 Thai	tis620_thai_ci	1
ucs2	UCS-2 Unicode	ucs2_general_ci	2
ujis	EUC-JP Japanese	ujis_japanese_ci	3
utf16	UTF-16 Unicode	utf16_general_ci	4
utf16le	UTF-16LE Unicode	utf16le_general_ci	4
utf32	UTF-32 Unicode	utf32_general_ci	4
utf8	UTF-8 Unicode	utf8_general_ci	3
utf8mb4	UTF-8 Unicode	utf8mb4_0900_ai_ci	4

问题所在的表的字符集是 utf8mb4 时，一个字符将占用 4 个字节。这意味着索引前缀最大长度为 3072 字节时，只能容纳 $3072 / 4 = 768$ 个字符。因此只要将上面建表语句索引字段的前缀长度设为 768 或者修改索引字段，让其小于 3072 字节。

解决方案

修改索引字段长度，即可成功创建索引。

9.1.3.4 RDS for MySQL 大小写参数敏感类问题

故障描述

用户将 RDS for 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。

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。
- 对于MySQL 5.7版本，支持在管理控制台或API创建数据库实例时指定表名大小写敏感，以及实例创建完成后设置表名大小写敏感（lower_case_table_names）。
- 对于MySQL 8.0版本，仅支持在管理控制台或API创建数据库实例时指定表名大小写敏感，创建完成的MySQL 8.0实例不支持设置表名大小写敏感（lower_case_table_names）。

9.1.3.5 RDS MySQL timeout 相关参数简介

MySQL中有多种timeout参数，RDS for MySQL也将相关参数提供给用户设置，如下表：

表 9-2 参数说明

参数名称	修改是否需要重启	参数含义
connect_timeout	否	控制客户端和MySQL服务端在建连接时，服务端等待三次握手成功的超时时间（秒），网络状态较差时，可以调大该参数。
idle_readonly_transaction_timeout	否	空闲的只读事务被kill前的等待时间，以秒为单位。（5.7.23版本之后支持）
idle_transaction_timeout	否	空闲事务被kill前的等待时间，以秒为单位。默认值设为0，代表永不kill。（5.7.23版本之后支持）
idle_write_transaction_timeout	否	空闲的读写事务被kill前的等待时间，以秒为单位。默认值设为0，代表永不kill。（5.7.23版本之后支持）
innodb_lock_wait_timeout	否	放弃事务前，InnoDB事务等待行锁的时间。

参数名称	修改是否需要重启	参数含义
innodb_rollback_on_timeout	是	innodb_rollback_on_timeout确定后，事务超时后InnoDB回滚完整的事务。
lock_wait_timeout	否	试图获得元数据锁的超时时间（秒）。
net_read_timeout	否	中止读数据之前从一个连接等待客户端网络包的秒数。
net_write_timeout	否	中止写数据之前等待一个网络包被写入TCP连接的秒数。
interactive_timeout	否	MySQL服务端在关闭交互式连接之前等待活动的秒数。
wait_timeout	否	MySQL服务端在关闭非交互式连接之前等待活动的秒数。

9.1.4 性能资源类

9.1.4.1 数据库磁盘满导致被设置 read_only

场景描述

业务侧出现如下报错：

```
The MySQL server is running with the --read-only option so it cannot execute this statement
```

原因分析

1. 进入实例详情页面，查看磁盘空间是否已满。
2. 登录数据库，查看read_only变量。
show variables like 'read_only';
3. 分析原因为实例磁盘空间满，数据库状态变更为只读，导致SQL语句执行失败。

解决方案

建议[扩容磁盘](#)。

9.1.4.2 Binlog 未清理导致磁盘占用高

场景描述

只读实例或主实例磁盘占用高，通过执行**SHOW BINARY LOGS**或**SHOW MASTER LOGS**，和其他实例对比发现，大量老的Binlog文件未被清理，导致磁盘占用很高。

原因分析

正常情况下，设置了[Binlog过期时间](#)，当Binlog备份至OBS，且超过过期时间后，会自动清理，如果长时间未清理，需考虑是否有其他复制异常因素导致。

排查思路：

1. [查看MySQL的错误日志](#)，查找是否有类似无法purge binlog的日志记录。
2022-01-18T05:39:03.139207+08:00 29 [Warning] file ./mysql-bin.106259 was not purged because it was being read by thread number 27490757
2. 分析是否有本地搭建复制关系、使用canal等工具监听该实例的Binlog，当主库未收到对应Binlog已被从库或工具获取的信息，会导致对应Binlog不被删除，导致Binlog积压。
3. 结合1中的异常binlog purge记录，分析本地从库或canal工具相应日志，排查网络状况等原因确认Binlog未被清理的原因。

解决方案

- 暂时停止该实例的其他Binlog监听任务，让该实例自动清理Binlog。
- 如果有本地从库，重新搭建复制关系。
- 如果使用canal等工具，重新建立Binlog拉取任务。

9.1.4.3 MySQL 只读实例磁盘占用远超主实例

场景描述

MySQL只读实例的磁盘占用比主实例高195GB。

原因分析

排查只读实例上运行的事务：

```
trx_waited_time: 0
***** 12. row *****
      trx_id: 422175775626376
      trx_state: RUNNING
      trx_started: 2022-07-22 11:00:26
      trx_requested_lock_id: NULL
      trx_wait_started: NULL
      trx_weight: 0
      trx_mysql_thread_id: 575709551
      trx_query: SELECT 'movie_id' FROM 'hg_app_pay_movie' WHERE 'member_id' = '12831953'
      trx_operation_state: Fetching rows
      trx_tables_in_use: 1
      trx_tables_locked: 0
      trx_lock_structs: 0
      trx_lock_memory_bytes: 1136
      trx_rows_locked: 0
      trx_rows_modified: 0
      trx_concurrency_tickets: 0
      trx_isolation_level: REPEATABLE READ
      trx_unique_checks: 1
      trx_foreign_key_checks: 1
      trx_last_foreign_key_error: NULL
      trx_adaptive_hash_latched: 0
      trx_adaptive_hash_timeout: 0
      trx_is_read_only: 1
      trx_autocommit_non_locking: 1
      trx_schedule_weight: NULL
      trx_waited_time: 0
***** 13. row *****
      trx_id: 422175775614616
      trx_state: RUNNING
      trx_started: 2022-07-21 15:27:39
      trx_requested_lock_id: NULL
      trx_wait_started: NULL
      trx_weight: 0
      trx_mysql_thread_id: 546262844
      trx_query: SELECT COUNT(*) AS tp_count FROM 'hg_member_history' 'a' LEFT JOIN 'hg_member_info' 'b' ON 'a`.`member_id`='b`.`id' LEFT JOIN 'hg_app_movie
      trx_operation_state: starting index read
      trx_tables_in_use: 3
      trx_tables_locked: 0
      trx_lock_structs: 0
      trx_lock_memory_bytes: 1136
      trx_rows_locked: 0
      trx_rows_modified: 0
      trx_concurrency_tickets: 0
```

发现有一直未提交的长事务，如上图所示（事务一天前开始），该长事务导致undo log一直未清理，累计了近一天的undo log，导致磁盘占用高。

解决方案

- 方式一：等待事务提交后，undo log会被清理，只读实例的磁盘占用恢复。
- 方式二：kill相应会话，停止长事务。

9.1.4.4 冷热数据问题导致 sql 执行速度慢

场景描述

从自建MySQL或友商MySQL迁移到云上RDS for MySQL实例，发现同一条sql语句执行性能远差于原数据库。

原因分析

同一条sql语句在数据库中执行第1次和第2次可能会性能差异巨大，这是由数据库的buffer_pool机制决定的：

- 第1次执行时，数据在磁盘上，称之为冷数据，读取需要一定的耗时。
- 读取完，数据会被存放于内存的buffer_pool中，称为热数据，读取迅速；对于热数据的访问速度极大的超过冷数据，所以当数据是热数据时，sql语句的执行速度会远快于冷数据。

该场景中，源端数据库中常用的数据一般是热数据，所以访问时速度极快。当数据迁移到云上RDS for MySQL时，第1次执行同样的sql语句，很可能是冷数据，就会访问较慢，但再次访问速度就会得到提升。

解决方案

该场景是正常现象，在同一个数据库中，我们经常会遇到第1次执行一条语句时很慢，但再次执行就很快，也是因为受到了buffer_pool的冷热数据原理的影响。

9.1.4.5 表空间膨胀问题

场景描述

在使用RDS for MySQL过程中，经常遇到表空间膨胀问题，例如：表中只有11774行数据，表空间却占用49.9GB，将该表导出到本地只有800M。

原因分析

场景1：DRS全量迁移阶段并行迁移导致

原因：DRS在全量迁移阶段，为了保证迁移性能和传输的稳定性，采用了行级并行的迁移方式。当源端数据紧凑情况下，通过DRS迁移到云上RDS for MySQL后，可能会出现数据膨胀现象，使得磁盘空间使用远大于源端。

场景2：大量删除操作后在表空间留下碎片所致

原因：当删除数据时，mysql并不会回收被删除数据占据的存储空间，而只做标记删除，尝试供后续复用，等新的数据来填补相应空间，如果一时半会，没有数据来填补这些空间，就造成了表空间膨胀，形成大量碎片；

可以通过如下SQL语句，查询某个表详细信息，DATA_FREE字段表示表空间碎片大小：

```
select * from information_schema.tables where table_schema='db_name' and table_name = 'table_name'\G
```

```
mysql> select * from information_schema.tables where table_schema='mall19wo' and table_name='deliveryman_track'\G
***** 1. row *****
TABLE_CATALOG: def
TABLE_SCHEMA: mall19wo
TABLE_NAME: deliveryman_track
TABLE_TYPE: BASE TABLE
ENGINE: InnoDB
VERSION: 10
ROW_FORMAT: Dynamic
TABLE_ROWS: 11968
AVG_ROW_LENGTH: 4479273
DATA_LENGTH: 53607940096
MAX_DATA_LENGTH: 0
INDEX_LENGTH: 802816
DATA_FREE: 54668558336
AUTO_INCREMENT: 94507
CREATE_TIME: 2022-06-28 23:39:00
UPDATE_TIME: 2022-07-07 11:03:22
CHECK_TIME: NULL
TABLE_COLLATION: utf8mb4_general_ci
CHECKSUM: NULL
CREATE_OPTIONS: row_format=DYNAMIC
TABLE_COMMENT: 骑手轨迹
1 row in set (0.00 sec)
```

解决方案

针对表空间膨胀的问题，可以进行表空间优化整理，从而缩小空间，执行如下SQL命令：

```
optimize table table_name;
```

注意：optimize table命令会有短暂锁表操作，所以进行表空间优化时，建议避开业务高峰期，避免影响正常业务的进行。

9.1.4.6 复杂查询造成磁盘满

场景描述

主机或只读实例偶尔出现磁盘占用高或磁盘占用满，其他只读实例或者备机磁盘空间占用正常。

原因分析

MySQL内部在执行复杂SQL时，会借助临时表进行分组（group by）、排序（order by）、去重（distinct）、Union等操作，当内存空间不够时，便会使用磁盘空间。

排查思路：

1. 因为其他只读实例和备机磁盘占用空间正常，且是偶尔出现，说明该实例磁盘占用高，与承载的业务相关。
2. 获取该实例的慢日志，分析磁盘占用高期间，是否有对应的慢SQL。
3. 如果有慢SQL，执行**explain [慢SQL语句]**，分析相应慢SQL语句。
4. 观察explain语句输出的extra列，是否有using temporary、using filesort，如果有，说明该语句用到了临时表或临时文件，数据量大的情况下，会导致磁盘占用高。

解决方案

1. 复杂查询语句导致磁盘打满，建议客户从业务侧优化响应查询语句，常见优化措施：
 - 加上合适的索引。

- 在where条件中过滤更多的数据。
 - 重写SQL，优化执行计划。
 - 如果不得不使用临时表，那么一定要减少并发度。
2. 临时规避措施：考虑业务侧优化复杂查询语句需要一定时间，可以通过临时扩容磁盘空间规避。

9.1.4.7 长事务导致规格变更或小版本升级失败

场景描述

长事务导致实例规格变更失败或小版本升级失败。

原因分析

- 规格变更过程或者小版本升级由于采用滚动执行的方式，最大程度减小对客户业务的影响，因此需要做主备切换。
- 主备切换时，为了保证数据一致性，需要先把主机设置readonly后，让主备的执行事务完全一致，才进行切换。
- 当主机上有长事务时，会导致主机设置readonly超时或失败，从而导致规格变更或小版本升级失败。

解决方案

1. 执行**show processlist**，查看正在执行的事务，使用如下命令，找到长事务。
select t.*,to_seconds(now())-to_seconds(t.trx_started) idle_time from INFORMATION_SCHEMA.INNODB_TRX t \G;

示例输出：

```
mysql> select t.*,to_seconds(now())-to_seconds(t.trx_started) idle_time from INFORMATION_SCHEMA.INNODB_TRX t \G
***** 1. row *****
      trx_id: 6168
      trx_state: RUNNING
      trx_started: 2021-09-16 11:08:27
      trx_requested_lock_id: NULL
      trx_wait_started: NULL
      trx_weight: 3
      trx_mysql_thread_id: 231
      trx_query: NULL
      trx_operation_state: NULL
      trx_tables_in_use: 0
      trx_tables_locked: 1
      trx_lock_structs: 3
      trx_lock_memory_bytes: 1136
      trx_rows_locked: 2
      trx_rows_modified: 0
      trx_concurrency_tickets: 0
      trx_isolation_level: REPEATABLE READ
      trx_unique_checks: 1
      trx_foreign_key_checks: 1
      trx_last_foreign_key_error: NULL
      trx_adaptive_hash_latched: 0
      trx_adaptive_hash_timeout: 0
      trx_is_read_only: 0
      trx_autocommit_non_locking: 0
      idle_time: 220
```

上述结果中idle_time是计算产生的，也是事务的持续时间。trx_mysql_thread_id是该事务的thread_id，和**show processlist**中的线程ID对应。

事务的trx_query是NULL，但不表示未执行事务，一个事务可能包含多个SQL，如果SQL执行完毕就不再显示。如果事务正在执行，InnoDB也无法得知该事务后续

还有没有SQL以及提交时间。此时，`trx_query`不能提供有意义的信息，因此为NULL。

2. kill长事务，再进行规格变更或小版本升级。
3. 建议客户在进行规格变更或小版本升级时，避免长事务执行。

9.1.4.8 RDS for MySQL 数据库报错 Native error 1461 的解决方案

场景描述

RDS for 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”。

9.1.4.9 RDS for MySQL 增加表字段后出现运行卡顿现象

故障描述

当给RDS for MySQL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。

9.1.4.10 长事务导致 UNDO 增多引起磁盘空间满

场景描述

实例触发磁盘满告警，一段时间后磁盘满告警自动恢复。

原因分析

- 由于MVCC机制，MySQL更新表中数据时会生成undo日志，会占用磁盘空间；所有会话的相关事务提交或回滚后，undo日志会被清理，导致磁盘空间下降。
- 当存在长事务时，长事务只要不提交，其他会话对相关表更新生成的undo就无法清理，导致磁盘空间一直上涨。

排查思路：

- 通过如下语句，检查是否有长时间不提交事务。

```
select t.*,to_seconds(now())-to_seconds(t.trx_started) idle_time from INFORMATION_SCHEMA.INNODB_TRX t \G;
```
- 通过审计日志或慢日志，检查是否存在大事务一次性插入大量数据。

解决方案

- kill相应的长事务。
- 建议业务侧避免在磁盘空间紧张时，执行长事务不提交，或执行大量插入。
- 提前进行磁盘扩容。

9.1.5 SQL 类

9.1.5.1 更新 emoji 表情数据报错 Error 1366

场景描述

业务插入或更新带有emoji表情的数据时，报错Error 1366。

```
java.sql.SQLException: Incorrect string value: '\xF0\x9F\x90\xB0\xE5\xA4...' for column 'username' at row 1 ;  
uncategorized SQLException for SQL []; SQL state [HY000]; error code [1366];  
Incorrect string value: '\xF0\x9F\x90\xB0\xE5\xA4...' for column 'username' at row 1;
```

原因分析

RDS for MySQL的字符集配置有误：

- emoji表情为特殊字符，需要4字节字符集存储。
- 该报错场景下MySQL字符集为utf-8，最多支持3个字节，需要修改为支持4个字节的字符集utf8mb4。

解决方案

1. 将存储emoji表情的字段的字符集修改为utf8mb4。
如果涉及的表和字段比较多，建议把对应表、数据库的编码也设置为utf8mb4。
参考命令：

```
ALTER DATABASE database_name CHARACTER SET= utf8mb4 COLLATE= utf8mb4_unicode_ci;  
ALTERTABLE table_name CONVERTTOCHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;  
ALTERTABLE table_name MODIFY 字段名 VARCHAR(128) CHARSET utf8mb4 COLLATE utf8mb4_unicode_ci;
```
2. 若对应字段的字符集已经是utf8mb4，则为客户端或MySQL服务端字符集转换问题，将客户端和MySQL服务端的字符集都设置为utf8mb4。

9.1.5.2 建表时 timestamp 字段默认值无效

场景描述

执行建表SQL语句失败，报错：ERROR 1067: Invalid default value for 'session_start'

```
CREATE TABLE cluster_membership
(
...
session_start TIMESTAMP DEFAULT '1970-01-01 00:00:01',
...
);
```

原因分析

关于timestamp字段：MySQL会把该字段插入的值从当前时区转换成UTC时间（世界标准时间）存储，查询时，又将其从UTC时间转化为当前时区时间返回。

timestamp类型字段的时间范围：'1970-01-01 00:00:01' UTC -- '2038-01-19 03:14:07' UTC，详见[官方文档](#)。

The **TIMESTAMP** data type is used for values that contain both date and time parts. **TIMESTAMP** has a range of '1970-01-01 00:00:01' UTC to '2038-01-19 03:14:07' UTC.

使用如下命令查看时区：

```
show variables like "%zone%";
```

由于使用的是UTC +8时区，所以timestamp字段默认值需要加8小时才是有效范围，即有效支持的范围是从1970-01-01 08:00:01开始。

```
mysql> show variables like "%zone%";
+-----+-----+
| Variable_name | Value |
+-----+-----+
| system_time_zone |      |
| time_zone       | +08:00 |
+-----+-----+
2 rows in set, 1 warning (0.00 sec)
```

解决方案

修改timestamp字段的默认值：

```
session_start TIMESTAMP DEFAULT '1970-01-01 08:00:01',
```

9.1.5.3 自增属性 AUTO_INCREMENT 为什么未在表结构中显示

场景描述

创建表时，添加了自增属性AUTO_INCREMENT，执行show create table，自增属性未在表结构中显示。

创建表：

```
mysql> CREATE TABLE xxx (
-> id int(10) NOT NULL AUTO_INCREMENT COMMENT '自增id',
-> account_id bigint(20) unsigned NOT NULL DEFAULT '0' COMMENT '关联account_base主键',
-> account_name varchar(128) NOT NULL DEFAULT '' COMMENT '用户名',
-> identity varchar(64) NOT NULL DEFAULT '' COMMENT '身份证',
-> mobile varchar(32) NOT NULL DEFAULT '' COMMENT '手机号',
-> score float(6,2) NOT NULL DEFAULT '0.00' COMMENT '分数',
-> utime bigint(20) unsigned NOT NULL DEFAULT '0' COMMENT '创建时间',
-> PRIMARY KEY (id),
-> UNIQUE KEY uique_index_account_id (account_id)
-> ) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8mb4
Query OK, 0 rows affected (0.01 sec)
```

执行show create table xxx，未显示自增属性AUTO_INCREMENT：

原因分析

经过排查，是因为参数“sql_mode”设置了NO_FIELD_OPTIONS属性。

sql_mode相关属性介绍：

- NO_FIELD_OPTIONS：不要在SHOW CREATE TABLE的输出中打印MySQL专用列选项。
- NO_KEY_OPTIONS：不要在SHOW CREATE TABLE的输出中打印MySQL专用索引选项。
- NO_TABLE_OPTIONS：不要在SHOW CREATE TABLE的输出中打印MySQL专用表选项（例如ENGINE）。

解决方案

将sql_mode的NO_FIELD_OPTIONS属性去掉即可。

9.1.5.4 存储过程和相关表字符集不一致导致执行缓慢

场景描述

RDS for MySQL存储过程执行很慢，处理少量数据耗时1min以上，而单独执行存储过程中的SQL语句却很快。

原因分析

存储过程和相关表、库的字符集不一致，导致查询结果存在大量字符转换，从而执行缓慢。

排查过程：

使用如下命令查看存储过程和相关表的定义，观察存储过程和表的字符集是否一致。

```
SHOW CREATE PROCEDURE xxx;
SHOW CREATE TABLE xxx
```

示例：

```
mysql> SHOW CREATE PROCEDURE testProc \G
***** 1. row *****
Procedure: showstuscore
sql_mode: STRICT_TRANS_TABLES,NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION
Create Procedure: xxx
```

```
character_set_client: utf8mb4
collation_connection: utf8mb4_general_ci
Database Collation: utf8_general_ci
1 row in set (0.01 sec)
```

可以看出，上述存储过程collation为utf8mb4_general_ci，而所在库collation默认为utf8_general_ci，collation值不一致，容易导致性能问题。

解决方案

将存储过程和相关表、库的字符集改成一致后，执行缓慢问题解决。

9.1.5.5 RDS MySQL 报错 ERROR [1412]的解决方法

场景描述

连接RDS MySQL执行SQL时，出现如下报错：

```
ERROR[1412]:Table definition has changed, please retry transaction``
```

原因分析

启动一致性快照事务后，其他会话（session）执行DDL语句导致。问题复现步骤：

1. 会话1启动一致性快照事务。

```
mysql> start transaction with consistent snapshot;
Query OK, 0 rows affected (0.00 sec)
```

2. 会话2执行DDL操作，修改表结构。

```
mysql> alter table t_sec_user add test int;
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

3. 会话1执行普通的查询语句。

```
mysql> select count(*) from t_sec_user;
ERROR 1412 (HY000): Table definition has changed, please retry transaction
mysql>
```

也可以通过Binlog或者审计日志，分析业务侧是否有同一个表DDL和一致性快照事务一起执行的情况。

解决方案

若经排查，是由上述原因引起的报错，需要业务侧避免同一个表的DDL语句和一致性快照事务同时执行。

9.1.5.6 创建二级索引报错 Too many keys specified

场景描述

创建二级索引失败，报错：Too many keys specified; max 64 keys allowed.

故障分析

MySQL对InnoDB每张表的二级索引的数量上限有限制，限制上限为64个，超过限制会报错“Too many keys specified; max 64 keys allowed”。详见[官方文档](#)。

[MySQL 8.0 Reference Manual / The InnoDB Storage Engine / InnoDB Limits](#)

15.22 InnoDB Limits

This section describes limits for `INNO_DB` tables, indexes, tablespaces, and other aspects of the `INNO_DB` storage engine.

- A table can contain a maximum of 1017 columns. Virtual generated columns are included in this limit.
- A table can contain a maximum of 64 secondary indexes.
- The index key prefix length limit is 3072 bytes for `INNO_DB` tables that use `DYNAMIC` or `COMPRESSED` row format.

解决方案

MySQL机制导致，建议优化业务，避免单表创建过多索引。

📖 说明

InnoDB表的其他限制：

1. 一个表最多可以包含1017列（包含虚拟生成列）。
2. InnoDB对于使用DYNAMIC或COMPRESSED行格式的表，索引键前缀长度限制为3072字节。
3. 多列索引最多允许16列，超过限制会报错。

9.1.5.7 存在外键的表删除问题

场景描述

删除MySQL表时，如果表中有外键（foreign key），会出现如下报错，且和用户权限无关：

```
ERROR 1451 (23000): Cannot delete or update parent row: a foreign key constraint fails .....
```

原因分析

这个表和其他表有外键关系，在MySQL中，设置了外键关联，会造成无法更新或删除数据，避免破坏外键的约束。

可以通过设置变量FOREIGN_KEY_CHECKS值为off，来关闭上述机制，详见[官方文档](#)。

解决方案

通过设置变量FOREIGN_KEY_CHECKS值为off，来关闭上述机制：

```
set session foreign_key_checks=off;  
drop table table_name;
```

9.1.5.8 distinct 与 group by 优化

场景描述

使用distinct或group by的语句执行比较慢。

原因分析

大部分情况下，distinct是可以转化成等价的group by语句。在MySQL中，distinct关键字的主要作用就是去重过滤。

distinct进行去重的原理是先进行分组操作，然后从每组数据中取一条返回给客户端，分组时有两种场景：

- distinct的字段全部包含于同一索引：该场景下MySQL直接使用索引对数据进行分组，然后从每组数据中取一条数据返回。
- distinct字段未全部包含于索引：该场景下索引不能满足去重分组需要，会用到临时表（首先将满足条件的数据写入临时表中，然后在临时表中对数据进行分组，返回合适的数据）。因为使用临时表会带来额外的开销，所以一般情况下性能会较差。

综上，在使用distinct或group by的时候，尽量在合理的情况下设置可以包含所有依赖字段的索引，优化示例：

- 没有合适索引，导致需要用到临时表。

```
mysql> show create table test;
+-----+
| Table | Create Table
+-----+
| test  | CREATE TABLE `test` (
  `id` int NOT NULL,
  `c1` int DEFAULT NULL,
  `c2` int DEFAULT NULL,
  `c3` int DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `c1` (`c1`),
  KEY `c2` (`c2`),
  KEY `c3` (`c3`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
+-----+
1 row in set (0.00 sec)

mysql> explain select distinct c1,c2,c3 from test;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE      | test  | NULL       | ALL | NULL         | NULL | NULL    | NULL | 1 | 100.00 | Using temporary |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set, 1 warning (0.00 sec)

mysql> explain select c1,c2,c3 from test group by c1,c2,c3;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE      | test  | NULL       | ALL | NULL         | NULL | NULL    | NULL | 1 | 100.00 | Using temporary |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set, 1 warning (0.03 sec)
```

- 有合适的索引，不会使用临时表，直接走索引。

```
mysql> alter table test add key(c1,c2,c3);
Query OK, 0 rows affected (0.10 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> show create table test;
+-----+-----+
| Table | Create Table |
+-----+-----+
| test  | CREATE TABLE `test` (
  `id` int NOT NULL,
  `c1` int DEFAULT NULL,
  `c2` int DEFAULT NULL,
  `c3` int DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `c1` (`c1`),
  KEY `c2` (`c2`),
  KEY `c3` (`c3`),
  KEY `c1_2` (`c1`,`c2`,`c3`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
+-----+-----+

mysql> explain select c1,c2,c3 from test group by c1,c2,c3;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE      | test | NULL        | index | c1_2          | c1_2 | 15      | NULL | 1 | 100.00 | Using index |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set, 1 warning (0.00 sec)

mysql> explain select distinct c1,c2,c3 from test;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | SIMPLE      | test | NULL        | index | c1_2          | c1_2 | 15      | NULL | 1 | 100.00 | Using index |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set, 1 warning (0.01 sec)
```

解决方案

在使用distinct或group by的时候，尽量在合理的情况下，创建可以包含所有依赖字段的索引。

9.1.5.9 字符集和字符序的默认选择方式

相关变量设置

参数组中默认character_set_server=utf8、collation_server=utf8_general_ci，可以在界面修改参数值。

默认选择方式

- 在创建数据库时，如果未显式指定库的字符集和字符序，则库的字符集和字符序采用character_set_server和collation_server参数的值；如果显式指定，则使用指定的字符集和字符序。
- 在创建数据表时，如果未显式指定表的字符集和字符序，则表默认字符集和字符序使用所在数据库的字符集和字符序；如果显式指定，则使用指定的字符集和字符序。
- 在创建数据表时，如果未显式指定字段的字符集和字符序，则字段使用所在表的字符集和字符序；如果显式指定，则使用指定的字符集和字符序。

示例1：不显式指定字符集、字符序的情况下创建数据库和数据表。

```
mysql> show variables like 'character_set_server';
+-----+
| Variable_name | Value |
+-----+
| character_set_server | utf8 |
+-----+
1 row in set (0.01 sec)

mysql> show variables like 'collation_server';
+-----+
| Variable name | Value |
+-----+
| collation_server | utf8_general_ci |
+-----+
1 row in set (0.01 sec)

mysql> create database test_default;
Query OK, 1 row affected (0.26 sec)

mysql> show create database test_default;
+-----+
| Database | Create Database |
+-----+
| test_default | CREATE DATABASE `test_default` /*!40100 DEFAULT CHARACTER SET utf8 */ /*!80016 DEFAULT ENCRYPTION='N' */ |
+-----+
1 row in set (0.00 sec)
```

```
mysql> create table test_default.t_default(name varchar(20));
Query OK, 0 rows affected (0.23 sec)

mysql> show create table test_default.t_default;
+-----+
| Table | Create Table |
+-----+
| t_default | CREATE TABLE `t_default` (
  `name` varchar(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
+-----+
1 row in set (0.01 sec)
```

示例2：显式指定库的字符集、字符序的情况下创建数据库。

```
mysql> create database test_define CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
Query OK, 1 row affected (0.00 sec)

mysql> show create database test_define;
+-----+
| Database | Create Database |
+-----+
| test_define | CREATE DATABASE `test_define` /*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci */ /*!80016 DEFAULT ENCRYPTION='N' */ |
+-----+
1 row in set (0.00 sec)

mysql> create table test_define.t_default(name varchar(20));
Query OK, 0 rows affected (0.08 sec)

mysql> show create table test_define.t_default;
+-----+
| Table | Create Table |
+-----+
| t_default | CREATE TABLE `t_default` (
  `name` varchar(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+-----+
1 row in set (0.01 sec)
```

示例3：显式指定表的字符集、字符序的情况下创建数据表。

```
mysql> create table test_define.t_define(name varchar(20) CHARACTER SET utf8 COLLATE utf8_bin);
Query OK, 0 rows affected, 2 warnings (0.05 sec)

mysql> show create table test_define.t_define;
+-----+
| Table | Create Table |
+-----+
| t_define | CREATE TABLE `t_define` (
  `name` varchar(20) COLLATE utf8_bin DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin |
+-----+
1 row in set (0.01 sec)
```

示例4：显式指定字段的字符集、字符序的情况下创建数据表。

```
mysql> create table test_define.t_v_define(name varchar(20) CHARACTER SET gbk COLLATE gbk_bin, str char(32) CHARACTER SET utf8 COLLATE utf8_bin);
Query OK, 0 rows affected, 2 warnings (0.06 sec)

mysql> show create table test_define.t_v_define;
+-----+
| Table | Create Table |
+-----+
| t_v_define | CREATE TABLE `t_v_define` (
  `name` varchar(20) CHARACTER SET gbk COLLATE gbk_bin DEFAULT NULL,
  `str` char(32) COLLATE utf8_bin DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin |
+-----+
1 row in set (0.01 sec)
```

9.1.5.10 MySQL 创建用户提示服务器错误

场景描述

用户账号在控制台界面上消失，创建不了同名账号，但使用账号名和旧密码还能连接。

创建用户失败的报错信息：

```
ERROR 1396 (HY000): Operation CREATE USER failed for xxx
```

原因分析

1. 查询确认，发现消失的账号在mysql.user表中已经被删除，因此在控制台不再显示。
2. 使用账号名和旧密码还能连接登录，说明使用的是**delete from mysql.user**方式删除用户。使用这种方式删除用户，需要执行**flush privileges**后，才会清理内存中相关数据，该用户才彻底不能登录。
3. 使用**delete from mysql.user**方式删除用户，无法重新创建相应账户（报错ERROR 1396），原因是内存中相关数据仍然存在。

```
mysql> CREATE USER 'test1'@'localhost' IDENTIFIED BY 'test1';
Query OK, 0 rows affected (0.03 sec)

mysql> DELETE FROM mysql.user WHERE Host='localhost'AND User='test1';
Query OK, 1 row affected (0.02 sec)

mysql> CREATE USER 'test1'@'localhost' IDENTIFIED BY 'test1';
ERROR 1396 (HY000): Operation CREATE USER failed for 'test1'@'localhost'
```

正确删除用户的方式为**drop user**语句，注意以下几点：

- **drop user**语句可用于删除一个或多个用户，并撤销其权限。
- 使用**drop user**语句必须拥有MySQL数据库的DELETE权限或全局CREATE USER权限。
- 在**drop user**语句的使用中，若没有明确地给出账户的主机名，则该主机名默认为“%”。

故障场景恢复示例：

创建用户后用**delete**删除用户，再创建同名用户时报错ERROR 1396。通过执行**flush privileges**后，可正常创建同名用户。

```
mysql> CREATE USER 'test1'@'localhost' IDENTIFIED BY 'test1';
ERROR 1396 (HY000): Operation CREATE USER failed for 'test1'@'localhost'
mysql> FLUSH HOSTS;
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE USER 'test1'@'localhost' IDENTIFIED BY 'test1';
ERROR 1396 (HY000): Operation CREATE USER failed for 'test1'@'localhost'
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE USER 'test1'@'localhost' IDENTIFIED BY 'test1';
Query OK, 0 rows affected (0.01 sec)
```

解决方案

- 方式一（推荐）：在业务低峰期，使用管理员账户执行**drop user user_name**删除用户，再重新创建该用户，修复该问题。
- 方式二：在业务低峰期，使用管理员账户执行**flush privileges**后，再重新创建该用户，修复该问题。建议开启数据库SQL审计日志，便于分析是哪个客户端删除了用户。

9.1.5.11 delete 大表数据后，再次查询同一张表时出现慢 SQL

场景描述

一次性删除多条宽列数据（每条记录数据长度在1GB左右），再次对同一张表进行增删改查时均执行缓慢，20分钟左右后恢复正常。

场景案例

1. 假定max_allowed_packet参数大小为1073741824。

2. 创建表。

```
CREATE TABLE IF NOT EXISTS zstest1
(
  id int PRIMARY KEY not null,
  c_longtext LONGTEXT
);
```

3. 向表中插入数据。

```
insert into zstest1 values(1, repeat('a', 1073741800));
insert into zstest1 values(2, repeat('a', 1073741800));
insert into zstest1 values(3, repeat('a', 1073741800));
insert into zstest1 values(4, repeat('a', 1073741800));
insert into zstest1 values(5, repeat('a', 1073741800));
insert into zstest1 values(6, repeat('a', 1073741800));
insert into zstest1 values(7, repeat('a', 1073741800));
insert into zstest1 values(8, repeat('a', 1073741800));
insert into zstest1 values(9, repeat('a', 1073741800));
insert into zstest1 values(10, repeat('a', 1073741800));
```

4. 删除数据。

```
delete from zstest1;
```

5. 执行查询语句。

```
select id from zstest1; //执行缓慢
```

原因分析

执行完delete操作后，后台purge线程会去清理标记为delete mark的记录。由于当前删除的数据量较大，purge遍历释放page的过程中会去获取page所在索引根节点的SX锁，导致select语句无法获取到根节点page的rw-lock，一直在等待。

解决方案

- 该场景为正常现象，等待purge操作完成后即可恢复正常。
- 扩大实例规格，提高purge效率。
- 调整优化业务，避免突然删除大量数据。如果需要删除表中所有数据，建议使用 **truncate table**。

9.1.5.12 为什么有时候用浮点数做等值比较查不到数据

原因分析

浮点数的等值比较问题是一种常见的浮点数问题。因为在计算机中，浮点数存储的是近似值而不是精确值，所以等值比较、数学运算等场景很容易出现预期外的情况。

MySQL中涉及浮点数的类型有float和double。如下示例中遇到的问题：

```
mysql> create table f(fnum float, dnum double);
Query OK, 0 rows affected (0.26 sec)

mysql> insert into f values(1.1, 1.2);
Query OK, 1 row affected (0.07 sec)

mysql> insert into f values(2.1, 2.2);
Query OK, 1 row affected (0.00 sec)

mysql> insert into f values(2.1, 3.2);
Query OK, 1 row affected (0.00 sec)

mysql> insert into f values(3.1, 3.2);
Query OK, 1 row affected (0.03 sec)

mysql> select * from f;
+-----+-----+
| fnum | dnum |
+-----+-----+
| 1.1 | 1.2 |
| 2.1 | 2.2 |
| 2.1 | 3.2 |
| 3.1 | 3.2 |
+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from f where fnum = 1.1;
Empty set (0.03 sec)

mysql> select * from f where fnum < 2;
+-----+-----+
| fnum | dnum |
+-----+-----+
| 1.1 | 1.2 |
+-----+-----+
1 row in set (0.00 sec)
```

解决方案

1. 使用精度的方法处理，使用字段与数值的差值的绝对值小于可接受的精度的方法。示例：

```
mysql> select * from f where fnum = 0.01;
Empty set (0.00 sec)

mysql> select * from f where abs(fnum - 1.1) < 0.01;
+-----+-----+
| fnum | dnum |
+-----+-----+
| 1.1 | 1.2 |
+-----+-----+
1 row in set (0.00 sec)
```

2. 使用定点数类型(DECIMAL)取代浮点数类型，示例：

```
mysql> create table d(d1 DECIMAL(5,2), d2 DECIMAL(5,2));
Query OK, 0 rows affected (0.09 sec)

mysql> insert into d values(1.1, 1.2);
Query OK, 1 row affected (0.02 sec)

mysql> insert into d values(2.1, 2.2);
Query OK, 1 row affected (0.01 sec)

mysql> insert into d values(3.1, 3.2);
Query OK, 1 row affected (0.01 sec)

mysql> select * from d;
+-----+-----+
| d1    | d2    |
+-----+-----+
| 1.10  | 1.20  |
| 2.10  | 2.20  |
| 3.10  | 3.20  |
+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from d where d1 = 1.1;
+-----+-----+
| d1    | d2    |
+-----+-----+
| 1.10  | 1.20  |
+-----+-----+
1 row in set (0.00 sec)
```

9.1.5.13 有外键的表无法删除报错 ERROR[1451]的解决方案

场景描述

数据库中的表，root用户也没有权限删除或修改。报错信息如下：

```
ERROR[1451] -Cannot deleteorupdatea parent row:  
aforeignkeyconstraintfails (...)
```

故障分析

该表的frm文件在sys_tables里也存在，这个表跟其他表有外键关系，因此不能直接删除。

因为RDS for MySQL中设置了foreign key关联，造成无法更新或删除数据，可以通过设置“FOREIGN_KEY_CHECKS”变量来避免这种情况。

解决方案

```
set session foreign_key_checks=off;  
drop table table_name;
```

关闭foreign_key_checks，即可删除表。

9.1.5.14 表字段类型转换失败的解决方法

场景描述

varchar字段使用char类型读取，不能用如下所示的方式转换：

故障分析

char()函数不能用于转换数据类型。

解决方案

RDS for MySQL的CAST()和CONVERT()函数可用来获取一个类型的值，并产生另一个类型的值。两者具体的语法如下：

```
CAST(value as type);  
CONVERT(value, type);
```

就是CAST(xxx AS 类型), CONVERT(xxx,类型)。

说明

可以转换的类型是有限制的。这个类型可以是以下值其中的一个：

- 二进制，同带binary前缀的效果：BINARY；
- 字符型，可带参数：CHAR()；
- 日期：DATE；
- 时间：TIME；
- 日期时间型：DATETIME；
- 浮点数：DECIMAL；
- 整数：SIGNED；
- 无符号整数：UNSIGNED。

9.1.5.15 RDS for MySQL 创建表失败报错 Row size too large 的解决方案

场景描述

RDS for 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。

9.1.5.16 RDS for MySQL 数据库报错 ERROR [1412]的解决方案

场景描述

用户使用时，出现如下报错信息：

ERROR[1412]:Table definition has changed, please retry transaction

问题可能出现的版本：MySQL-5.7.31.2

故障分析

原因一：启动一致性快照事务引起。

场景1：

```
mysql> start transaction with consistent snapshot;  
Query OK, 0 rows affected (0.00 sec)
```

场景2：

```
mysql> alter table t_sec_user add test int;  
Query OK, 0 rows affected (0.01 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

场景3：

```
mysql> select count(*) from t_sec_user;  
ERROR 1412 (HY000): Table definition has changed, please retry transaction  
mysql>
```

原因二：对binlog进行了DDL类操作。

```
# at 13793936  
#210325 18:44:06 server id 1605802263 end_log_pos 13794065 CRC32 0x4df2db06 Query thread_id=19314888 exec_time=1 error_code=0  
use `zzkj`/*!*/;  
SET TIME_STAMP=1616669046/*!*/;  
SET @session.sql_mode=1346371584/*!*/;  
/*!C utf8mb4 *//*!*/;  
SET @session.character_set_client=45,@session.collation_connection=45,@session.collation_server=33/*!*/;  
ALTER TABLE bas_cabinet_box ADD lock_status tinyint(4)  
/*!*/;
```

解决方案

若经排查，是由上述原因引起的报错，请客户从业务侧解决问题。

9.1.6 连接类

9.1.6.1 连接数据库报错 Access denied

场景描述

客户端连接数据库异常，返回错误：Error 1045: Access denied for user xxx

处理方法

1. 连接了错误的主机

问题原因：业务连接了错误的数据库主机，该主机上相应用户或客户端IP没有权限访问。

解决方案：仔细检查要连接的数据库主机名，确保正确。

2. 用户不存在

问题原因：客户端连接时，使用的用户不存在。

解决方案：

- 使用管理员账户登录数据库，执行如下命令检查目标用户是否存在。

```
SELECT User FROM mysql.user WHERE User='xxx';
```

- 如果用户不存在，创建相应用户。

```
CREATE USER 'xxx'@'xxxxxx' IDENTIFIED BY 'xxxx';
```

3. 用户存在，但客户端IP无访问权限

问题原因：客户端使用的用户存在，但是客户端IP没有该数据库的访问权限。

解决方案：

- 使用管理员账户登录数据库，执行如下命令，检查目标用户允许哪些客户端IP连接。

```
SELECT Host, User FROM mysql.user WHERE User='xxx';
```

- 如果上述查询出的Host不包含客户端IP所在网段，则需要赋予相应访问权限。例如，赋予test用户192.168.0网段访问权限。

```
GRANT ALL PRIVILEGES ON *.* TO 'root'@'192.168.0.%' IDENTIFIED BY 'password' WITH GRANT OPTION;  
FLUSH PRIVILEGES;
```

4. 密码错误

问题原因：用户对应的密码错误，或忘记密码

解决方案：

- 确定目标密码是否错误，由于密码用于身份验证，因此无法从MySQL以明文形式读取用户密码，但可以将密码的哈希字符串与目标密码的

“PASSWORD”函数值进行比较，确定目标密码是否正确，示例SQL语句：

```
mysql> SELECT Host, User, authentication_string, PASSWORD('12345') FROM mysql.user  
WHERE User='test';
```

```
+-----+-----+-----+-----+  
| Host | User | authentication_string | PASSWORD('12345') |  
+-----+-----+-----+-----+  
| % | test | *6A23DC5E7446019DC9C1778554ED87BE6BA61041 |  
*00A51F3F48415C7D4E8908980D443C29C69B60C9 |  
+-----+-----+-----+-----+  
2 rows in set, 1 warning (0.00 sec)
```

从上面例子可以看出，PASSWORD('12345')的哈希值与 authentication_string列不匹配，这表明目标密码“12345”是错误的。

- 如果需要重置用户密码，参考如下SQL语句：

```
set password for 'test'@'%' = 'new_password';
```

5. 密码包含特殊字符被Bash转义

问题原因：Linux默认的Bash环境下，使用命令行连接数据库，用户密码中包含特殊字符会被环境转义，导致密码失效。

例如，在Bash环境下，用户test的密码为“test\$123”，使用命令mysql -hxxx -u test -ptest\$123，连接数据库会报错ERROR 1045 (28000): Access denied。

解决方案：通过用单引号将密码括起来，防止Bash解释特殊字符。

```
mysql -hxxx -u test -p'test$123'
```

6. 用户设置了REQUIRE SSL，但客户端使用非SSL连接

排查思路：

- 排查报错用户名是否强制使用SSL连接，执行：`show create user 'xxx'`，如果出现“REQUIRE SSL”属性，说明该用户必须使用SSL连接。
- 排查是否使用过如下类似语句给用户授权。
`GRANT ALL PRIVILEGES ON . TO 'ssluser'@'localhost' IDENTIFIED BY 'zdh1234' REQUIRE SSL;`
- 检查目标用户的`ssl_type`值，如果不为空，说明该用户需要使用SSL连接。
`SELECT User, Host, ssl_type FROM mysql.user WHERE User='xxx';`

解决方案：

- 客户端使用SSL方式数据库连接，请参考[SSL连接方式](#)。
- 去除用户SSL连接权限，参考命令：`ALTER USER 'test'@'xxxxx' REQUIRE NONE;`

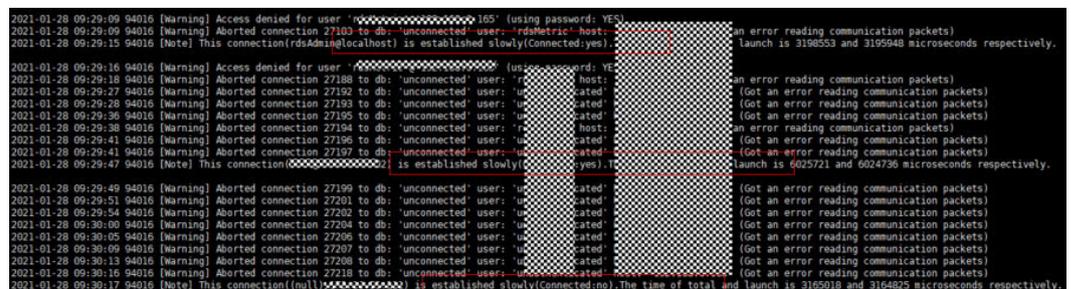
9.1.6.2 RDS for MySQL 建立连接慢导致客户端超时报 connection established slowly

场景描述

业务在高峰期时，客户端经常出现向MySQL建立连接超时，导致系统登录需要十几秒。

原因分析

1. [查看MySQL的错误日志](#)，观察是否有如下信息：`connection xxx is established slowly`。示例：



```
2021-01-28 09:29:09 94016 [Warning] Access denied for user 'root@165' (using password: YES)
2021-01-28 09:29:09 94016 [Warning] Aborted connection 27193 to db: 'unconnected' user: 'rdametric' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:15 94016 [Note] This connection('rdametric@localhost') is established slowly(connected=yes). The time of total and launch is 3198553 and 3195948 microseconds respectively.
2021-01-28 09:29:16 94016 [Warning] Access denied for user 'root@165' (using password: YES)
2021-01-28 09:29:18 94016 [Warning] Aborted connection 27188 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:27 94016 [Warning] Aborted connection 27192 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:28 94016 [Warning] Aborted connection 27193 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:36 94016 [Warning] Aborted connection 27195 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:38 94016 [Warning] Aborted connection 27194 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:41 94016 [Warning] Aborted connection 27196 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:41 94016 [Warning] Aborted connection 27197 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:47 94016 [Note] This connection('root@165') is established slowly(connected=yes). The time of total and launch is 6025721 and 6024736 microseconds respectively.
2021-01-28 09:29:49 94016 [Warning] Aborted connection 27199 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:51 94016 [Warning] Aborted connection 27201 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:29:54 94016 [Warning] Aborted connection 27202 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:00 94016 [Warning] Aborted connection 27204 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:05 94016 [Warning] Aborted connection 27206 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:06 94016 [Warning] Aborted connection 27207 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:13 94016 [Warning] Aborted connection 27208 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:16 94016 [Warning] Aborted connection 27210 to db: 'unconnected' user: 'root' host: '10.10.10.1' (Get an error reading communication packets)
2021-01-28 09:30:17 94016 [Note] This connection('root@165') is established slowly(connected=no). The time of total and launch is 3165018 and 3164825 microseconds respectively.
```

有上述日志，说明存在某些连接超过一定时间仍未被MySQL处理，客户端的超时时间大于该时间，就会报错。

2. 进一步查看线程池配置（默认开启），可以在控制台查看。

可以看出，`threadpool_size`为1，`threadpool_stall_limit`为500ms，`threadpool_oversubscribe`为3，线程池处理连接等待的时间主要与上述3个参数相关：

- 当线程池所有线程在忙碌，线程池中的调度线程会每隔500ms（`threadpool_stall_limit`）创建一个新线程。此时，每个线程组平均每500ms才能处理一个新的连接。如果队列太长，很可能导致客户端超时。
- 所有线程都在忙碌是指工作线程达到线程池总线程数，在大量建立连接时，总线程数计算方法为：`threadpool_size*(threadpool_oversubscribe+1)`

3. 因此，`threadpool_size`参数值设置不合理或过小，会导致业务高峰期，新连接建立慢。

解决方案

对于存在大量新建连接，建议调大`threadpool_size`和`threadpool_oversubscribe`增加线程总数，调小`threadpool_stall_limit`，减少新建连接等待。

减少线程重复创建与销毁部分的开销，提高性能，同时它也限制了MySQL的`runing`线程数，关键时刻可以保护系统，防止雪崩。

正常情况下，线程池适用于大量短连接的场景，如果客户是长连接，并且连接数量不多（客户端使用了连接池等情况），线程池的作用不大，此时调整`threadpool_size`和`threadpool_oversubscribe`两个参数扩大线程总数，或者直接关闭线程池。

9.1.6.3 客户端 TLS 版本与 RDS for MySQL 不一致导致 SSL 连接失败

场景描述

某业务客户端连接到云上RDS for MySQL失败，但是连接到自建环境或其他环境可以成功，均使用了SSL连接。

原因分析

排查步骤：

1. [查看MySQL的错误日志](#)，观察到如下报错：

```
2021-07-09T10:30:58.476586+08:00 212539 [Warning] SSL errno: 337678594, SSL errmsg:
error:14209102:SSL routines:tls_early_post_process_client_hello:unsupported
protocol2021-07-09T10:30:58.476647+08:00 212539 [Note] Bad
handshake2021-07-09T10:32:43.535738+08:00 212631 [Warning] SSL errno: 337678594, SSL errmsg:
error:14209102:SSL routines:tls_early_post_process_client_hello:unsupported
protocol2021-07-09T10:32:43.535787+08:00 212631 [Note] Bad
handshake2021-07-09T10:50:03.401100+08:00 213499 [Warning] SSL errno: 337678594, SSL errmsg:
error:14209102:SSL routines:tls_early_post_process_client_hello:unsupported
protocol2021-07-09T10:50:03.401161+08:00 213499 [Note] Bad
handshake2021-07-09T10:53:44.458404+08:00 213688 [Warning] SSL errno: 337678594, SSL errmsg:
error:14209102:SSL routines:tls_early_post_process_client_hello:unsupported
protocol2021-07-09T10:53:44.458475+08:00 213688 [Note] Bad handshake
```

2. 从报错信息`unsupported protocol`可以看出，很可能和TLS版本相关，使用如下命令，分别查看RDS for MySQL和自建MySQL的TLS版本。

```
show variables like '%tls_version%';
```

发现RDS for MySQL为TLS v1.2版本，自建MySQL为TLS v1.1版本，存在差异。进一步确认客户端TLS版本，与自建MySQL一致，因此出现连接自建MySQL成功，连接云上RDS for MySQL失败。

解决方案

客户端升级TLS版本到TLS v1.2。

如果使用官方JDBC驱动`mysql-connector/J`，可参考[官方文档](#)，配置方法：

TLS versions: The allowable versions of TLS protocol can be restricted using the connection properties `tlsVersions` and, for X DevAPI connections and for release 8.0.19 and later, `xdevapi.tls-versions` (when `xdevapi.tls-versions` is not specified, it takes up the value of `tlsVersions`). If no such restrictions have been specified, Connector/J attempts to connect to the server with the TLSv1.2 and TLSv1.3.

9.1.6.4 RDS for MySQL 客户端连接实例后会自动断开

故障描述

RDS for 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RDS for 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RDS for 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，RDS for 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

9.1.6.5 RDS for 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 'Test1i@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RDS for 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，RDS for 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@'%'这个账号所对应的密码，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账号。

----结束

故障二

步骤1 排查是否在创建RDS for 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。

步骤2 重置root账号密码。

```
ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY 'XXX';  
flush privileges;
```

```
mysql> ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY '1234567890';  
Query OK, 0 rows affected (0.01 sec)  
mysql> flush privileges;  
Query OK, 0 rows affected (0.01 sec)
```

修改后用户就可以正常登录root账号。

----结束

9.1.6.7 RDS for 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

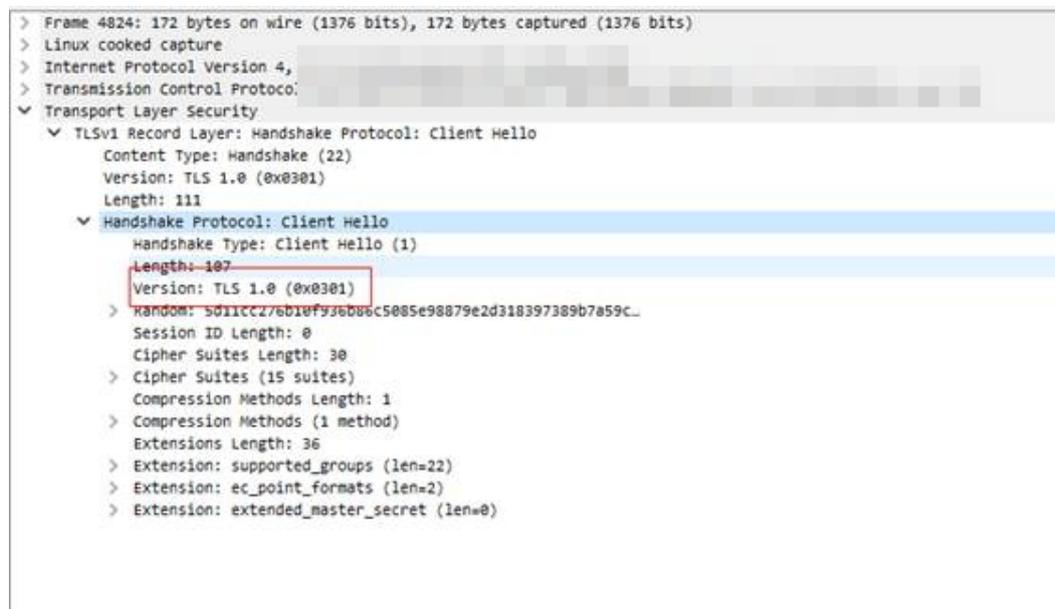
用户新建实例，用代码连接该数据库时出现报错：

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

RDS for 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9-2：

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 9-2 连接失败抓包结果

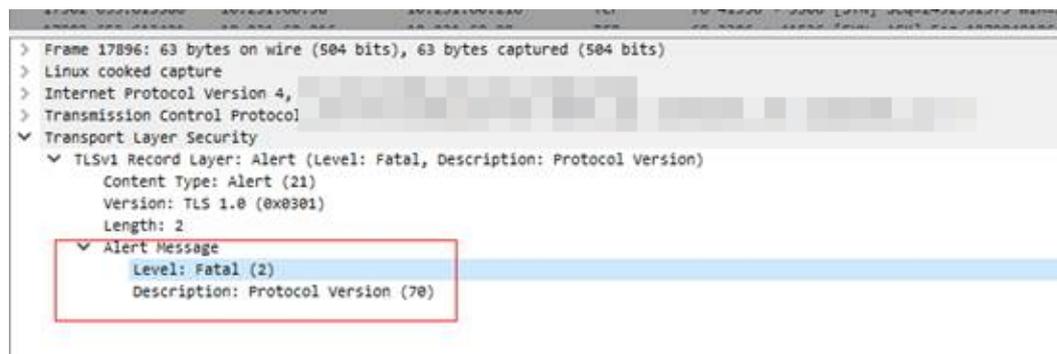


```
> Frame 4824: 172 bytes on wire (1376 bits), 172 bytes captured (1376 bits)  
> Linux cooked capture  
> Internet Protocol Version 4,  
> Transmission Control Protocol:  
▼ Transport Layer Security  
  ▼ TLSv1 Record Layer: Handshake Protocol: Client Hello  
    Content Type: Handshake (22)  
    Version: TLS 1.0 (0x0301)  
    Length: 111  
      ▼ Handshake Protocol: Client Hello  
        Handshake Type: Client Hello (1)  
        Length: 107  
        Version: TLS 1.0 (0x0301)  
        > Random: 5D11CC276B10F936B86C5085E98879E2D318397389B7A59C...  
        Session ID Length: 0  
        Cipher Suites Length: 30  
        Cipher Suites (15 suites)  
        Compression Methods Length: 1  
        Compression Methods (1 method)  
        Extensions Length: 36  
        > Extension: supported_groups (len=22)  
        > Extension: ec_point_formats (len=2)  
        > Extension: extended_master_secret (len=0)
```

故障分析

从MySQL-server的回复中如图9-3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。

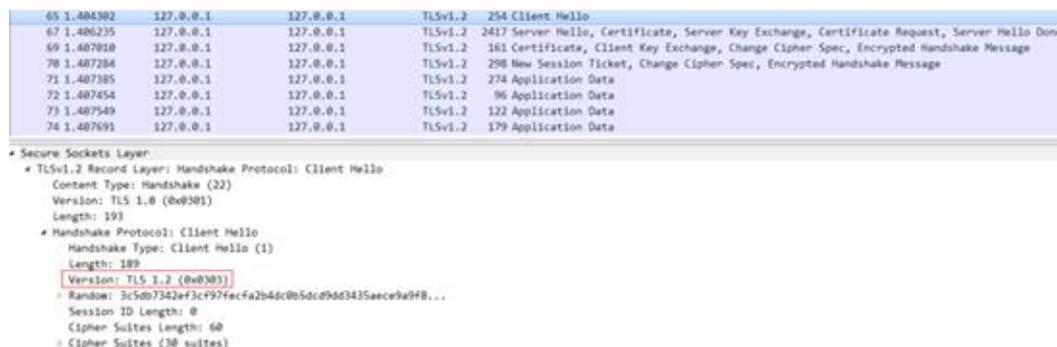
图 9-3 MySQL-server 的回复



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9-4，可以正常连接的客户端支持TLS1.2，并支持30个密码套件。

图 9-4 正常连接抓包结果



9.1.6.8 客户端超时参数设置不当导致连接超时退出

场景描述

使用数据库时，经常遇到连接退出，导致后续语句执行失败的情况。

原因分析

在使用连接器或API连接数据库时，客户端会有一些默认的参数配置。其中有一些比较重要的参数如socketTimeout、connectTimeout等，会影响客户端连接的超时时间。如果超过这个时间，一直没使用的连接就会断开。

解决方案

- 将socketTimeout、connectTimeout等参数的默认值调整为合适的值。
- 在程序中注意处理断线重连的功能。
- 推荐直接使用连接池。

9.1.6.9 RDS for MySQL 在启用了 SSL 验证连接功能后，导致代码（php/java/python）等连接数据库失败

场景描述

用户开启了SSL验证连接功能，使用代码执行SQL连接数据库出现如下报错：

图 9-5 连接失败

```
[2019-07-19 11:24:44] local.ERROR: SQLSTATE[HY000] [2026] SSL connection error: unknown error number (SQL: update `on_server_user`_100038 set `last_login_time` = 1563506684 where `game_id` = 1 and `user_device` = 60645783-1676-4A09-88F1-23C2A9720256) [] exception="Object() Illuminate\Database\QueryException(code: 2026): SQLSTATE[HY000] [2026] SSL connection error: unknown error number (SQL: update `on_server_user`_100038 set `last_login_time` = 1563506684 where `game_id` = 1 and `user_device` = 60645783-1676-4A09-88F1-23C2A9720256) at /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Connection.php:684 #00exception(code: 2026): SQLSTATE[HY000] [2026] SSL connection error: unknown error number at /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Connectors/Connector.php:69 [stacktrace] #0 /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Connection.php(624): Illuminate\Database\Connection->runQueryCallback('update `on_serv...', Array, Object(Closure)) #1 /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Connection.php(490): Illuminate\Database\Connection->run('update `on_serv...', Array, Object(Closure)) #2 /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Connection.php(423): Illuminate\Database\Connection->affectingStatement('update `on_serv...', Array) #3 /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Query/Builder.php(928): Illuminate\Database\Connection->update('update `on_serv...', Array) #4 /data/web/laravel/center.wxgame.youx1765.com/vendor/laravel/framework/src/Illuminate/Database/Eloquent/Builder.php(780): Illuminate\Database\Query\Builder->update(Array) #5 /data/web/laravel/center.wxgame.youx1765.com/app/Http/Models/OnServerUserModel.php(49): Illuminate\Database\Eloquent\Builder->update(Array) #6 /data/web/laravel/center.wxgame.youx1765.com/app/Services/OnServerService.php(202): App\Http\Models\OnServerUserModel->updateData(1, 100038, Array, Array) #7 /data/web/laravel/center.wxgame.youx1765.com/app/Services/MiniGameService.php(113): App\Services\OnServerService->updateUserLogin(1, 100038, 60645783-1676-4...)
```

故障分析

用户开启了SSL连接功能，则只能使用SSL连接方式。

解决方案

开启SSL验证连接功能，使用SSL连接方式连接数据库，请参考[SSL连接方式](#)。

9.1.7 其他使用问题

9.1.7.1 自增字段取值

RDS for MySQL对自增字段赋值有以下几种方法：

```
# 表结构
CREATE TABLE animals (
  id MEDIUMINT NOT NULL AUTO_INCREMENT,
  name CHAR(30) NOT NULL,
  PRIMARY KEY (id)
);
```

1. 不对自增字段赋值，数据库会自动将自增值填入字段中。AUTO_INCREMENT为自增。

```
mysql> INSERT INTO animals (name) VALUES ('fish'),('cat'),('penguin'),('lax'),('whale'),('ostrich');
Query OK, 6 rows affected (0.01 sec)
Records: 6 Duplicates: 0 Warnings: 0
mysql> select * from animals;
```

```
+----+-----+
| id | name |
+----+-----+
| 1 | fish |
| 2 | cat |
| 3 | penguin |
| 4 | lax |
| 5 | whale |
| 6 | ostrich |
+----+-----+
```

```
6 rows in set (0.00 sec)
```

```
mysql> show create table animals;
```

```
+-----+-----+
| Table | Create Table |
+-----+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 |
+-----+-----+
```

2. 对自增字段赋0或null值，数据库会自动将自增值填入字段中。
AUTO_INCREMENT为自增。

```
mysql> INSERT INTO animals (id,name) VALUES(0,'groundhog');
Query OK, 1 row affected (0.00 sec)
mysql> INSERT INTO animals (id,name) VALUES(NULL,'squirrel');
Query OK, 1 row affected (0.01 sec)
mysql> select * from animals;
+----+-----+
| id | name   |
+----+-----+
| 1 | fish   |
| 2 | cat    |
| 3 | penguin |
| 4 | lax    |
| 5 | whale  |
| 6 | ostrich |
| 7 | groundhog |
| 8 | squirrel |
+----+-----+8
rows in set (0.00 sec)
mysql> show create table animals;
+-----+-----+
| Table | Create Table |
+-----+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=utf8 |
+-----+-----+
```

3. 直接使用大于AUTO_INCREMENT的值X，数据库会将X填入字段并修改
AUTO_INCREMENT=X+1。

```
mysql> INSERT INTO animals (id,name) VALUES(100,'rabbit');
Query OK, 1 row affected (0.00 sec)
mysql> select * from animals;
+----+-----+
| id | name   |
+----+-----+
| 1 | fish   |
| 2 | cat    |
| 3 | penguin |
| 4 | lax    |
| 5 | whale  |
| 6 | ostrich |
| 7 | groundhog |
| 8 | squirrel |
| 100 | rabbit |
+----+-----+
9 rows in set (0.00 sec)
mysql> show create table animals;
+-----+-----+
| Table | Create Table |
+-----+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=101 DEFAULT CHARSET=utf8 |
+-----+-----+
```

4. 直接使用小于AUTO_INCREMENT但不冲突的值。数据可以插入，但
AUTO_INCREMENT不变。

```
mysql> INSERT INTO animals (id,name) VALUES(50,'middle');
Query OK, 1 row affected (0.00 sec)
mysql> select * from animals;
+----+-----+
| id | name   |
+----+-----+
| 1 | fish   |
| 2 | cat    |
| 3 | penguin |
| 4 | lax    |
+----+-----+
```

```
| 5 | whale |
| 6 | ostrich |
| 7 | groundhog |
| 8 | squirrel |
| 50 | middle |
| 100 | rabbit |
+-----+
10 rows in set (0.00 sec)
mysql> show create table animals;
+-----+
| Table | Create Table |
+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT, `name`
char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=101 DEFAULT
CHARSET=utf8 |
+-----+
```

5. 直接使用负值。数据可以插入，但AUTO_INCREMENT不变。

```
mysql> INSERT INTO animals (id,name) VALUES(-50,'-middle');
Query OK, 1 row affected (0.00 sec)
mysql> select * from animals;
+-----+
| id | name |
+-----+
|-50 | -middle |
| 1 | fish |
| 2 | cat |
| 3 | penguin |
| 4 | lax |
| 5 | whale |
| 6 | ostrich |
| 7 | groundhog |
| 8 | squirrel |
| 50 | middle |
| 100 | rabbit |
+-----+
11 rows in set (0.00 sec)
mysql> show create table animals;
+-----+
| Table | Create Table |
+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT,
`name` char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=101 DEFAULT
CHARSET=utf8 |
+-----+
```

9.1.7.2 表的自增 AUTO_INCREMENT 初值与步长

AUTO_INCREMENT的初值与步长由“auto_increment_increment”和“auto_increment_offset”参数决定。

- auto_increment_offset: AUTO_INCREMENT值的初值。
- auto_increment_increment: AUTO_INCREMENT值每次增长的步长。
- 当 auto_increment_offset > auto_increment_increment 时，实际使用时初值会变为为auto_increment_increment。
- 当 auto_increment_offset <= auto_increment_increment 时，自增值计算方式：
值 = auto_increment_offset + N*auto_increment_increment (N为插入的数据条数)

在RDS for MySQL中“auto_increment_increment”和“auto_increment_offset”参数默认都为1，如需修改请在控制台修改，具体操作请参见[修改RDS for MySQL实例参数](#)。

举例：

- auto_increment_offset=1, auto_increment_increment=1, 那么初值为1, 步长为1。

```
mysql> show variables like 'auto_inc%';
+-----+-----+
| Variable_name      | Value |
+-----+-----+
| auto_increment_increment | 1 |
| auto_increment_offset  | 1 |
+-----+-----+
mysql> create table auto_test1(id int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`));
Query OK, 0 rows affected (0.09 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table      | Create Table
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
`id` int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT
CHARSET=utf8 |
+-----+-----+
mysql> insert into auto_test1 values(0), (0), (0);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
+----+
3 rows in set (0.01 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table      | Create Table
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
`id` int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=4
DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
```

- 修改auto_increment_increment=2, 步长变为2。

```
mysql> set session auto_increment_offset=2;
Query OK, 0 rows affected (0.02 sec)
mysql> show variables like 'auto_inc%';
+-----+-----+
| Variable_name      | Value |
+-----+-----+
| auto_increment_increment | 2 |
| auto_increment_offset  | 1 |
+-----+-----+
mysql> insert into auto_test1 values(0), (0), (0);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
| 4 |
| 6 |
| 8 |
+----+
6 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table      | Create Table
+-----+-----+
```

- ```
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
 `id` int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`))
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.01 sec)
```
- auto\_increment\_offset=10, auto\_increment\_increment=2, 初值为2（因为 auto\_increment\_offset > auto\_increment\_increment），步长为2。  
mysql> set session auto\_increment\_offset=10;  
mysql> set session auto\_increment\_increment=2;  
mysql> show variables like 'auto\_inc%';  
+-----+-----+  
| Variable\_name | Value |  
+-----+-----+  
| auto\_increment\_increment | 2 |  
| auto\_increment\_offset | 10 |  
+-----+-----+  
mysql> create table auto\_test2(id int NOT NULL AUTO\_INCREMENT, PRIMARY KEY (`id`)); Query OK,  
0 rows affected (0.08 sec)  
mysql> show create table auto\_test2;  
+-----+  
-----+  
| Table | Create Table |  
+-----+  
-----+  
| auto\_test2 | CREATE TABLE `auto\_test2` ( `id` int NOT NULL AUTO\_INCREMENT, PRIMARY KEY  
(`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 |  
+-----+  
-----+  
1 row in set (0.01 sec)  
mysql> insert into auto\_test2 values(0), (0), (0);  
Query OK, 3 rows affected (0.00 sec)  
Records: 3 Duplicates: 0 Warnings: 0  
mysql> select \* from auto\_test2;  
+----+  
| id |  
+----+  
| 2 |  
| 4 |  
| 6 |  
+----+  
3 rows in set (0.01 sec)  
mysql> show create table auto\_test2;  
+-----+  
-----+  
| Table | Create Table |  
+-----+  
-----+  
| auto\_test2 | CREATE TABLE `auto\_test2` (   
`id` int NOT NULL AUTO\_INCREMENT, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO\_INCREMENT=8  
DEFAULT CHARSET=utf8 |  
+-----+  
-----+
  - auto\_increment\_offset=5, auto\_increment\_increment=10, 初值为5, 步长为10。  
mysql> set session auto\_increment\_offset=5; mysql> set session auto\_increment\_increment=10;  
mysql> show variables like 'auto\_inc%';  
+-----+-----+  
| Variable\_name | Value |  
+-----+-----+  
| auto\_increment\_increment | 10 |  
| auto\_increment\_offset | 5 |  
+-----+-----+  
mysql> create table auto\_test3(id int NOT NULL AUTO\_INCREMENT, PRIMARY KEY (`id`));  
mysql> show create table auto\_test3;  
+-----+  
-----+

```
| Table | Create Table |
+-----+
+-----+
+-----+
| auto_test3 | CREATE TABLE `auto_test3` (`id` int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 |
+-----+
+-----+
mysql> insert into auto_test3 values(0), (0), (0);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test3;
+----+
| id |
+----+
| 5 |
| 15 |
| 25 |
+----+
mysql> show create table auto_test3;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test3 | CREATE TABLE `auto_test3` (
`id` int NOT NULL AUTO_INCREMENT, PRIMARY KEY (`id`))
) ENGINE=InnoDB AUTO_INCREMENT=35 DEFAULT CHARSET=utf8 |
+-----+-----+
```

### 9.1.7.3 自增字段值跳变的原因

出现表中的自增字段取值不连续的情况，可能原因有以下几种：

- 初值与步长问题，步长不为1会导致自增字段取值不连续。

```
mysql> show variables like 'auto_inc%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| auto_increment_increment | 2 |
| auto_increment_offset | 1 |
+-----+-----+
mysql> select * from auto_test1;
+----+
| id |
+----+
| 2 |
| 4 |
| 6 |
| 8 |
+----+
```

- 直接修改表的AUTO\_INCREMENT，会导致自增字段取值跳变。

```
mysql> select * from animals;
+----+-----+
| id | name |
+----+-----+
1	fish
2	cat
3	penguin
+----+-----+	
mysql> show create table animals;	
+-----+-----+	
Table	Create Table
+-----+-----+	
animals	CREATE TABLE `animals` (
`id` mediumint NOT NULL AUTO_INCREMENT,
`name` char(30) NOT NULL,
PRIMARY KEY (`id`))
ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 |
+-----+-----+
```

```
+-----+-----+
mysql> alter table animals AUTO_INCREMENT=100;
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table animals;
+-----+-----+
| Table | Create Table |
+-----+-----+
| animals | CREATE TABLE `animals` (
`id` mediumint NOT NULL AUTO_INCREMENT,
`name` char(30) NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=100 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> INSERT INTO animals (id,name) VALUES(0,'rabbit');
Query OK, 1 row affected (0.00 sec)
mysql> select * from animals;
+----+-----+
| id | name |
+----+-----+
1	fish
2	cat
3	penguin
100	rabbit
+----+-----+
9 rows in set (0.00 sec)
```

- 插入数据时直接指定自增字段的取值，会导致自增字段取值跳变。

```
mysql> select * from animals;
+----+-----+
| id | name |
+----+-----+
1	fish
2	cat
3	penguin
+----+-----+	
mysql> INSERT INTO animals (id,name) VALUES(100,'rabbit');	
Query OK, 1 row affected (0.00 sec)	
mysql> select * from animals;	
+----+-----+	
id	name
+----+-----+	
1	fish
2	cat
3	penguin
100	rabbit
+----+-----+
9 rows in set (0.00 sec)
```

- 未提交的事务或回滚的事务，会导致AUTO\_INCREMENT增长，但回滚后不会下降。后续如果再次插入数据就会导致数据中的自增字段发生跳变。

```
mysql> show create table auto_test1;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
`id` int NOT NULL AUTO_INCREMENT,
PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
+----+
mysql> begin;
```

```
Query OK, 0 rows affected (0.02 sec)
mysql> insert into auto_test1 values (0),(0),(0);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
+----+
6 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test1 |
CREATE TABLE `auto_test1` (
 `id` int NOT NULL AUTO_INCREMENT,
 PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> rollback;
Query OK, 0 rows affected (0.05 sec)
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
+----+
3 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
 `id` int NOT NULL AUTO_INCREMENT,
 PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> insert into auto_test1 values (0),(0),(0);
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
| 7 |
| 8 |
| 9 |
+----+
6 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
 `id` int NOT NULL AUTO_INCREMENT,
 PRIMARY KEY (`id`)
```

- ```
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8 |
+-----+-----+
• 数据插入后，AUTO_INCREMENT变化，然后删除对应的数据行，
  AUTO_INCREMENT不会下降，后续如果再次插入数据就会导致数据中的自增字
  段发生跳变。
mysql> show create table auto_test1;
+-----+-----+
| Table   | Create Table                               |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
  `id` int NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
+----+
mysql> insert into auto_test1 values (0),(0),(0);
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
+----+
6 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table   | Create Table                               |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
  `id` int NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> delete from auto_test1 where id>3;
mysql> select * from auto_test1;
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
+----+
3 rows in set (0.00 sec)
mysql> show create table auto_test1;
+-----+-----+
| Table   | Create Table                               |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
  `id` int NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> insert into auto_test1 values (0),(0),(0);
```

```
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test1;
```

```
+----+
| id |
+----+
| 1 |
| 2 |
| 3 |
| 7 |
| 8 |
| 9 |
+----+
```

```
6 rows in set (0.00 sec)
```

```
mysql> show create table auto_test1;
```

```
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test1 | CREATE TABLE `auto_test1` (
`id` int NOT NULL AUTO_INCREMENT,
PRIMARY KEY (`id`))
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8 |
+-----+-----+
```

- 因为一些原因（比如唯一键冲突），使得插入数据最终未成功的，有可能导致 AUTO_INCREMENT 跳变。

```
mysql> create table auto_test7(`id` int NOT NULL AUTO_INCREMENT, cred_id int UNIQUE, PRIMARY KEY (`id`));
```

```
Query OK, 0 rows affected (0.64 sec)
```

```
mysql> insert into auto_test7 values(null, 1);
```

```
Query OK, 1 row affected (0.03 sec)
```

```
mysql> show create table auto_test7;
```

```
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test7 | CREATE TABLE `auto_test7` ( `id` int NOT NULL AUTO_INCREMENT, `cred_id` int
DEFAULT NULL, PRIMARY KEY (`id`), UNIQUE KEY `cred_id` (`cred_id`)) ENGINE=InnoDB
AUTO_INCREMENT=2 DEFAULT CHARSET=utf8 |
+-----+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> insert into auto_test7 values(null, 1);
```

```
ERROR 1062 (23000): Duplicate entry '1' for key 'auto_test7.cred_id'
```

```
mysql> show create table auto_test7;
```

```
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test7 | CREATE TABLE `auto_test7` ( `id` int NOT NULL AUTO_INCREMENT, `cred_id` int
DEFAULT NULL, PRIMARY KEY (`id`), UNIQUE KEY `cred_id` (`cred_id`)) ENGINE=InnoDB
AUTO_INCREMENT=3 DEFAULT CHARSET=utf8 |
+-----+-----+
```

- 批量插入数据时（如insert...select、load file等），自增键的申请是分批申请的，每批申请2的n次方个序号，用完继续申请，没用完也不会退回，所以可能会导致 AUTO_INCREMENT 跳变。

```
mysql> create table auto_test5_tmp(id tinyint not null AUTO_INCREMENT, name varchar(8), PRIMARY KEY (`id`));
```

```
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> select * from auto_test5;
```

```
+-----+-----+
| id | name |
+-----+-----+
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | X |
| 5 | Y |
| 6 | Z |
| 8 | A |
| 9 | B |
+-----+-----+
```

```
| 10 | C |
| 11 | X |
| 12 | Y |
| 13 | Z |
+----+-----+
12 rows in set (0.00 sec)
mysql> insert into auto_test5_tmp select 0,name from auto_test5;
Query OK, 12 rows affected (0.01 sec)
Records: 12 Duplicates: 0 Warnings: 0
mysql> select * from auto_test5_tmp;
+----+-----+
| id | name |
+----+-----+
| 1 | A |
| 2 | B |
| 3 | C |
| 4 | X |
| 5 | Y |
| 6 | Z |
| 7 | A |
| 8 | B |
| 9 | C |
| 10 | X |
| 11 | Y |
| 12 | Z |
+----+-----+
12 rows in set (0.00 sec)
mysql> show create table auto_test5_tmp;
+-----+-----+
| Table | Create Table |
+-----+-----+
| auto_test5_tmp | CREATE TABLE `auto_test5_tmp` ( `id` tinyint NOT NULL AUTO_INCREMENT, `name` varchar(8) DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=16 DEFAULT CHARSET=utf8 |
+-----+-----+
```

9.1.7.4 修改表的自增 AUTO_INCREMENT 值

修改方法如下：

- 当 AUTO_INCREMENT 大于表中数据的最大值时，可以在取值范围内任意修改为更大的值。

```
mysql> show create table animals;
+-----+-----+
| Table | Create Table |
+-----+-----+
| animals | CREATE TABLE `animals` ( `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=101 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> select * from animals;
+----+-----+
| id | name |
+----+-----+
| -50 | -middle |
| 1 | fish |
| 2 | cat |
| 50 | middle |
| 100 | rabbit |
+----+-----+
11 rows in set (0.00 sec)
mysql> alter table animals AUTO_INCREMENT=200;
Query OK, 0 rows affected (0.22 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table animals;
+-----+-----+
| Table | Create Table |
+-----+-----+
```

```
+-----+-----+
| animals | CREATE TABLE `animals` (
| `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL,
| PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=200 DEFAULT CHARSET=utf8 |
+-----+-----+
● 当AUTO_INCREMENT大于表中数据的最大值时，如果修改后的指定值仍大于数
  据的最大值，则修改为指定值成功。否则，默认会修改为数据最大值+1。
mysql> select * from animals;
+----+-----+
| id | name  |
+----+-----+
| -50 | -middle |
| 1 | fish  |
| 2 | cat   |
| 50 | middle |
| 100 | rabbit |
+----+-----+
mysql> show create table animals;
+-----+-----+
| Table | Create Table
+-----+-----+
| animals | CREATE TABLE `animals` (
| `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL,
| PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=200 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> alter table animals AUTO_INCREMENT=150;
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table animals;
+-----+-----+
| Table | Create Table
+-----+-----+
| animals | CREATE TABLE `animals` (
| `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL,
| PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=150 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> alter table animals AUTO_INCREMENT=50;
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table animals;
+-----+-----+
| Table | Create Table
+-----+-----+
| animals | CREATE TABLE `animals` (
| `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL,
| PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=101 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> delete from animals where id=100;
Query OK, 1 row affected (0.00 sec)
mysql> select * from animals;
+----+-----+
| id | name  |
+----+-----+
| -50 | -middle |
| 1 | fish  |
| 2 | cat   |
| 50 | middle |
+----+-----+
10 rows in set (0.00 sec)
mysql> alter table animals AUTO_INCREMENT=50;
Query OK, 0 rows affected (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table animals;
+-----+-----+
| Table | Create Table
+-----+-----+
| animals | CREATE TABLE `animals` (
| `id` mediumint NOT NULL AUTO_INCREMENT, `name` char(30) NOT NULL,
| PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=51 DEFAULT CHARSET=utf8 |
+-----+-----+
```

```
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

- **AUTO_INCREMENT无法修改为负数。**
mysql> alter table animals AUTO_INCREMENT=-1;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '-1' at line 1

9.1.7.5 自增主键达到上限，无法插入数据

场景现象

插入数据时报错：ERROR 1062 (23000): Duplicate entry 'xxx' for key 'xxx'

原因分析

自增主键的字段取值达到上限，无法继续增长，导致新插入的数据生成的自增主键值与表中上一条数据相同，因为自增主键的值不可重复，插入失败报错。

```
mysql> create table auto_test5(id tinyint not null AUTO_INCREMENT, name varchar(8), PRIMARY KEY
(`id`));
Query OK, 0 rows affected (0.06 sec) mysql> insert into auto_test5(name) values('A'),('B'),('C');
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test5;
+----+-----+
| id | name |
+----+-----+
| 1 | A |
| 2 | B |
| 3 | C |
+----+-----+
3 rows in set (0.00 sec)
mysql> alter table auto_test5 AUTO_INCREMENT=125;
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0
mysql> show create table auto_test5;
+-----+-----+-----+-----+
| Table | Create Table |
+-----+-----+-----+-----+
| auto_test5 | CREATE TABLE `auto_test5` (
`id` tinyint NOT NULL AUTO_INCREMENT, `name` varchar(8) DEFAULT NULL,
PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=125 DEFAULT CHARSET=utf8 |
+-----+-----+-----+-----+
mysql> insert into auto_test5(name) values('X'),('Y'),('Z');
Query OK, 3 rows affected (0.00 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> select * from auto_test5;
+----+-----+
| id | name |
+----+-----+
| 1 | A |
| 2 | B |
| 3 | C |
| 125 | X |
| 126 | Y |
| 127 | Z |
+----+-----+
6 rows in set (0.00 sec)
mysql> show create table auto_test5;
+-----+-----+-----+-----+
| Table | Create Table |
+-----+-----+-----+-----+
| auto_test5 | CREATE TABLE `auto_test5` ( `id` tinyint NOT NULL AUTO_INCREMENT, `name` varchar(8)
DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO_INCREMENT=127 DEFAULT CHARSET=utf8 |
+-----+-----+-----+-----+
```



```
`id` tinyint NOT NULL AUTO_INCREMENT, `name` varchar(8) DEFAULT NULL,
PRIMARY KEY (`id`) ) ENGINE=InnoDB AUTO_INCREMENT=127 DEFAULT CHARSET=utf8 |
+-----+-----+
mysql> alter table auto_test6 modify column id int NOT NULL AUTO_INCREMENT;
Query OK, 6 rows affected (0.15 sec)
Records: 6 Duplicates: 0 Warnings: 0
mysql> show create table auto_test6;
+-----+-----+
| Table   | Create Table                                     |
+-----+-----+
| auto_test6 | CREATE TABLE `auto_test6` (
  `id` int NOT NULL AUTO_INCREMENT, `name` varchar(8) DEFAULT NULL,
  PRIMARY KEY (`id`) ) ENGINE=InnoDB AUTO_INCREMENT=128 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.00 sec)
mysql> insert into auto_test6(name) values('D');
Query OK, 1 row affected (0.01 sec)
mysql> select * from auto_test6;
+----+-----+
| id | name |
+----+-----+
| 1  | A    |
| 2  | B    |
| 3  | C    |
| 125 | X    |
| 126 | Y    |
| 127 | Z    |
| 128 | D    |
+----+-----+
7 rows in set (0.00 sec)
mysql> show create table auto_test6;
+-----+-----+
| Table   | Create Table                                     |
+-----+-----+
| auto_test6 | CREATE TABLE `auto_test6` (
  `id` int NOT NULL AUTO_INCREMENT, `name` varchar(8) DEFAULT NULL,
  PRIMARY KEY (`id`) ) ENGINE=InnoDB AUTO_INCREMENT=129 DEFAULT CHARSET=utf8 |
+-----+-----+
1 row in set (0.01 sec)
```

9.1.7.6 空用户的危害

MySQL中是允许用户名为 " " 的用户存在，本章节介绍数据库中存在这种空用户时的危害。

MySQL中使用空用户时，它将可以匹配任何用户名。这一特性也会带来多种安全性、功能性危害。所以，在实际使用过程中应避免使用空用户。

- 安全性危害

- 当存在空用户时，连接时可以使用任意用户名进行登录。
- 如果空用户有密码，则使用任意用户名和空用户的密码即可登录数据库，并获得空用户所拥有的所有权限。示例：

#没有空用户时，使用非法用户名 'abcd'，连接失败

```
mysql> select user,host from mysql.user;
```

```
+-----+-----+
| user      | host      |
+-----+-----+
| root      | %         |
| mysql.infoschema | localhost |
| mysql.session | localhost |
| mysql.sys  | localhost |
+-----+-----+
```

```
mysql -uabcd -h127.0.0.1 -P3306 -pTest_1234
```

```
mysql: [Warning] Using a password on the command line interface can be insecure.
ERROR 1045 (28000): Access denied for user 'abcd'@'localhost' (using password: YES)
```

```
# 创建空用户后，使用非法用户名 'abcd'，密码用空用户的密码，连接成功
mysql> create user ''@'localhost' IDENTIFIED BY 'Test_1234';
mysql> select user,host from mysql.user;
+-----+-----+
| user      | host      |
+-----+-----+
| root      | %         |
|           | localhost |
| mysql.infoschema | localhost |
| mysql.session | localhost |
| mysql.sys   | localhost |
+-----+-----+
mysql -uabcd -h127.0.0.1 -P3306 -pTest_1234
mysql: [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 37Server version: 8.0.22-debug Source distribution
Copyright (c) 2000, 2020, 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>
```

- 如果空用户没有密码，则使用任意用户名即可免密登录数据库，并获得空用户所拥有的所有权限。示例：

```
#存在无密码的空用户时，可以使用任意用户免密登录数据库。
```

```
mysql> create user ''@'localhost';
Query OK, 0 rows affected (8.87 sec)
mysql> select user,host from mysql.user;
+-----+-----+
| user      | host      |
+-----+-----+
| root      | %         |
|           | localhost |
| mysql.infoschema | localhost |
| mysql.session | localhost |
| mysql.sys   | localhost |
+-----+-----+
mysql -uabcd -h127.0.0.1 -P3306
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 39Server version: 8.0.22-debug Source distribution
Copyright (c) 2000, 2020, 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>
#-----
mysql -usdhsjkdshk -h127.0.0.1 -P3306
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 40Server version: 8.0.22-debug Source distribution
Copyright (c) 2000, 2020, 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>
```

- 功能性危害

当存在空用户时，可能因为匹配出错，导致正常的用户名无法登录。

示例：存在空用户与root用户的host有重叠时，导致root用户无法使用密码登录，或者使用空用户的密码登录后无法进入root的权限。

```
mysql> create user ''@'localhost';
Query OK, 0 rows affected (8.87 sec)
mysql> select user,host from mysql.user;
+-----+-----+
| user      | host      |
+-----+-----+
| root      | %         |
|           | localhost |
| mysql.infoschema | localhost |
+-----+-----+
```

```
| mysql.session | localhost |
| mysql.sys    | localhost |
+-----+-----+
# 用root的密码无法登录
mysql -uroot -h127.0.0.1 -P3306 -pTest_root
mysql: [Warning] Using a password on the command line interface can be insecure.
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: YES)
# 用空用户的密码(免密)登录后实际是空用户登录, 没有root权限。
mysql -uroot -h127.0.0.1 -P3306
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 45Server version: 8.0.22-debug Source distribution
Copyright (c) 2000, 2020, 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> select user,host from mysql.user;
ERROR 1142 (42000): SELECT command denied to user ''@'localhost' for table 'user'
mysql>
```

9.1.7.7 pt-osc 工具连接 RDS for MySQL 主备实例卡住

场景描述

使用pt-osc工具（pt-online-schema-change）进行online DDL操作时，会遇到连接本地单机实例执行命令成功，但连接云上RDS for MySQL主备实例会卡住的情况，如下图所示，没有输出：



原因分析

pt-osc的工作原理：

1. 创建一个与原表结构相同的空表，表名是 `_new` 后缀。
2. 修改1创建的空表的表结构。
3. 在原表上加三个触发器：delete/update/insert，用于复制数据过程中，将原表中要执行的语句在新表中执行。
4. 将原表数据以数据块（chunk）的形式复制到新表。
5. **rename**原表为旧表，并把新表**rename**为原表名，然后删除旧表。
6. 删除触发器。

因为涉及大量数据复制，对于有从库的MySQL，必然会带来主备复制延迟，可能影响从库业务。考虑从库延迟情况，pt-osc工具提供以下几个控制选项：

- `--max-lag`
- `--check-interval`
- `--recursion-method`
- `--check-slave-lag`

因此，从库延迟超过max-lag，则停止复制数据，等待check-interval秒后再开始复制数据；check-slave-lag指定slave的机器，只会对比这台slave的延迟情况。recursion-method是主库寻找从库的方法：processlist（默认值，关注从库延迟）、hosts、dsn、none（忽略从库延迟）。更多信息，请参见[pt-osc官方文档](#)。

本案例场景中：

- pt-osc工具连接云上RDS for MySQL主备实例卡住，是因为存在主从复制延迟导致工具停止复制数据。配置项“--recursion-method=none”表示忽略主从延迟，添加该配置项可以解决问题。
- 忽略主从延迟，会导致复制数据比较快，当需要尽可能的对服务产生小的影响，可以设置“--max-load”配置项。

解决方案

pt-osc工具加上“--recursion-method=none”配置项，忽略复制延迟，即可解决问题。

pt-osc常见命令参考：

- 增加字段

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "ADD COLUMN content text" D=aaa,t=tmp_test --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --print --execute
```
- 删除字段

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "DROP COLUMN content " D=aaa,t=tmp_test --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --quiet --execute
```
- 修改字段

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "MODIFY COLUMN age TINYINT NOT NULL DEFAULT 0" D=aaa,t=tmp_test --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --quiet --execute
```
- 字段改名

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "CHANGE COLUMN age address varchar(30)" D=aaa,t=tmp_test --no-check-alter --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --quiet --execute
```
- 增加索引

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "ADD INDEX idx_address(address)" D=aaa,t=tmp_test --no-check-alter --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --print --execute
```
- 删除索引

```
pt-online-schema-change --user=root --password=xxx --host=xxx --alter "DROP INDEX idx_address" D=aaa,t=tmp_test --no-check-alter --no-check-replication-filters --alter-foreign-keys-method=auto --recursion-method=none --print --execute
```

如果业务需要关注复制延迟，可以根据业务需要调整如下参数：max-lag、check-interval、recursion-method、check-slave-lag。更多信息，请参见[pt-osc官方文档](#)。

9.2 RDS for SQL Server

9.2.1 RDS for SQL Server 数据库恢复到新实例后新建用户，提示用户已存在的解决方法

故障描述

SQL Server数据库通过备份文件恢复到新实例，新建用户，进行数据库用户绑定时失败。提示“User, group, or role 'ecology' already exists in the current database”。

原因分析

数据库在备份恢复后，该用户已存在，只是界面上未显示。

解决方案

以rdsuser用户执行以下命令关联，更换对应库的用户名，以ecology为例。

```
use [DBName];  
Exec sp_change_users_login 'AUTO_FIX','ecology';
```