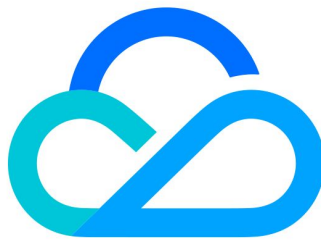


云数据库 PostgreSQL

内核能力介绍

产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

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

【商标声明】

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

【服务声明】

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

文档目录

内核能力介绍

- 内核版本概述

- 内核版本更新动态

- 查看内核版本

- 自研内核功能介绍

 - 逻辑复制槽故障转移 (Failover Slot)

 - TDE 透明数据加密

 - 异步 invalid buffer

 - 进程监控

 - 使用 starocks_fdw 实现读写分离

内核能力介绍

内核版本概述

最近更新时间：2024-04-03 16:44:22

本文为您介绍云数据库 PostgreSQL 内核版本的背景信息和优势。

背景信息

PostgreSQL（简称 PG）是一款全球流行的企业级开源数据库，2017、2018连续两年蝉联 DB-Engine 年度数据库，2019年荣获 OSCON（The O'Reilly Open Source Convention）开源软件终身成就奖，2020年再次获得年度数据库冠军。

腾讯云数据库 PostgreSQL 是经过腾讯多年的业务沉淀和积累，在社区版本的 PostgreSQL 之上，根据不同的业务特征加入了特色功能和优化的内核版本。其不仅对 PostgreSQL 数据库引擎、权限管理、复制性能等方面进行了大量优化，同时提升了在云上的易用性和可维护性。

内核版本号

内核版本号由两部分组成：**社区版本号**和**内部版本号**。

```
v{社区主版本号.社区次版本号}_r{不兼容版本号.兼容版本号}
如：v12.7_r1.1
```

v 代表 Version，r 代表 Release，如：v12.7_r1.1对应的社区版本是12.7，r1.1是代表腾讯云迭代版本，由 `r {不兼容版本号} . {兼容版本号}` 这两部分构成。

说明：

Version：PostgreSQL 社区版本号，与社区对应。包含有两个数字，分别是 Major Version、Minor Version；Major Version 从10开始（9.3、9.4、9.5、9.6按照旧递进关系），Minor Version 从1开始。当社区次要版本 released 后，腾讯云会定时跟随社区版本进行版本更新。

Release：云数据库 PostgreSQL 迭代用的版本号，表达云数据库对 PostgreSQL 所做的修改。包含有两个数字，分别是 Incompatible Version 和 Compatible Version；Incompatible Version 从1开始，Compatible Version 从0开始。

不兼容版本号

不兼容定义：用新的数据库内核程序启动数据库实例，若启动失败或功能无法正常使用，该修改就是不兼容的。当内核支持了不兼容的功能，需要将不兼容版本号增加1。不兼容版本号将用于判断能否直接进行二进制小版本升级。

兼容性版本号

增加兼容性的修改时，需要将兼容版本号增加1。

版本支持列表

PostgreSQL 10

PostgreSQL 11

PostgreSQL 12

PostgreSQL 13

PostgreSQL 14

PostgreSQL 15

PostgreSQL 16

您可以通过 [PostgreSQL 控制台](#) 实例列表，查看云数据库 PostgreSQL 的数据库版本。不同的版本号代表对应兼容的社区版本，如：PostgreSQL 12 兼容社区 PostgreSQL 12.x 的所有子版本，12.1、12.2、12.3等。版本生命周期请参考 [大版本生命周期说明](#)。

版本规则

Version + Release 一起组成完整版本，称为数据库内核版本。

Major Version 对应一个社区演化分支，各自迭代，如10、11、12等，分别拥有自己的版本号序列。

当社区宣布 Major Version 不再维护时，相应版本将停止新购。但腾讯云会一直维护此版本的数据库实例，直到用户实例主动销毁或迁移。

Minor Version 跟随社区的变更，Release 根据腾讯云数据库内核所做功能的变更而变更。

优势

相比开源版本 PostgreSQL 具有如下特点：

更强的稳定性

云数据库 PostgreSQL 在开源 PostgreSQL 的基础上，优化了部分数据库底层实现，使数据库因外界影响受到崩溃的问题极大减少，提升了数据库的稳定性。

主从同步性能增强

社区版 PostgreSQL 在大量 DDL 处理时，主从同步性能变得极其差，云数据库 PostgreSQL 在此方面进行极大优化，整体提升高达3万倍。

更短的访问延迟

云数据库 PostgreSQL 在大量短连接场景进行了建连优化，使数据库连接性能消耗减少80%，极大提升了短连接场景的性能体验。

内核版本更新动态

最近更新时间：2024-10-28 14:55:21

本文为您介绍云数据库 PostgreSQL 内核版本更新动态。如需升级内核小版本，请参考 [升级内核小版本](#)；如需升级大版本，请参考 [升级数据库大版本](#)。

PostgreSQL 16

V16.2_r1.6

cos_fdw 增加安全鉴权功能，限制访问同账户购买的 COS。

V16.2_r1.5

修复审计极速版影响行数一直显示为 0 的问题；
修复审计精细版扩展协议下 SELECT 语句无法审计的问题；
支持 SQL 限流插件 tencentdb_sql_throttling；
修复社区 PG 逻辑复制 BUG：大量 update 导致逻辑复制分析wal日志速度慢；

V16.2_r1.4

修复 zhparser 权限问题；
修复 vacuum 陷入死循环问题。

V16.2_r1.3

兼容社区版本16.2；
支持 database 级别 CPU 隔离。

V16.0_r1.2

修复审计日志长连接导致的内存泄露；
修复 dblink 非法参数导致 core；
出于安全考虑，普通日志记录 COPY 语句；
tencentdb_superuser 支持创建 replication origin 权限；
修复 cos_fdw 进行大分区表扫描时的超时报错问题。

V16.0_r1.1

修复 CVE-2023-5869

V16.0_r1.0

兼容社区16.0版本。

PostgreSQL 15

V15.6_r1.12

cos_fdw 增加安全鉴权功能，限制访问同账户购买的 COS。

V15.6_r1.11

修复 zhparser 权限问题；

修复审计极速版影响行数一直显示为 0 的问题；

修复审计精细版扩展协议下 SELECT 语句无法审计的问题；

支持 sql 限流插件 tencentdb_sql_throttling；

支持 pldebugger 插件。

V15.6_r1.10

兼容社区版本15.6；

支持 database 级别 CPU 隔离。

V15.1_r1.9

修复审计日志长连接导致的内存泄露；

修复 dblink 非法参数导致 core；

出于安全考虑，普通日志记录 COPY 语句；

tencentdb_superuser 支持创建 replication origin 权限；

修复 cos_fdw 进行大分区表扫描时的超时报错问题。

V15.1_r1.8

修复 CVE-2023-5869。

V15.1_r1.7

审计日志性能优化。

修复 failover slot 慢于备机 checkpoint 位点时，从备机发起备份失败的问题。

V15.1_r1.6

支持大版本升级。

更新部分插件版本。

V15.1_r1.5

兼容社区15.1版本。

修复 CVE-2023-39417 安全漏洞。

支持 tencentdb_system_stat 进程监控插件。

支持 pg_squeeze、pg_similarity 插件。

V15.1_r1.4

兼容社区15.1版本。

支持 dblink 跨实例访问。

rds_server_handler 插件支持使用命令清空所有 cache。

支持数据库全审计。

禁止 COPY PROGRAM 行为。

支持 tencentdb_superuser 插件。

V15.1_r1.3

兼容社区15.1版本。

支持 pgvector 插件。

V15.1_r1.2

兼容社区15.1版本。

对部分插件在创建和升级时可能造成的安全问题进行加固。

V15.1_r1.0

兼容社区15.1版本。

新增 tencentdb_superuser 权限，允许创建 event_trigger。

PostgreSQL 14

V14.11_r1.23

cos_fdw 增加安全鉴权功能，限制访问同账户购买的 COS。

V14.11_r1.22

修复 zhparser 权限问题；

修复审计极速版影响行数一直显示为 0 的问题；

修复审计精细版扩展协议下 SELECT 语句无法审计的问题；

支持 sql 限流插件 tencentdb_sql_throttling。

V14.11_r1.21

兼容社区14.11版本；
支持数据库对象 database 级别 CPU 资源隔离。

V14.2_r1.18

修复审计日志长连接导致的内存泄露；
修复 dblink 非法参数导致 core；
出于安全考虑，普通日志记录 COPY 语句；
tencentdb_superuser 支持创建 replication origin 权限；
修复 cos_fdw 进行大分区表扫描时的超时报错问题。
社区 BUG 修复：备机 promote 后时间线回切导致数据损坏。

V14.2_r1.16

修复 CVE-2023-5869。

V14.2_r1.15

审计日志性能优化；

V14.2_r1.14

支持大版本升级；
更新部分插件版本。

V14.2_r1.12

兼容社区14.2版本。
修复 CVE-2023-39417 安全漏洞。
支持 tencentdb_system_stat 进程监控插件。
支持 pg_squeeze、pg_similarity 插件。

V14.2_r1.12

修复 CVE-2023-39417 安全漏洞；
支持 tencentdb_system_stat 进程监控插件；
支持 pg_squeeze、pg_similarity 插件；

V14.2_r1.11

支持 dblink 跨实例访问；
rds_server_handler 插件支持使用命令清空所有 cache；
支持审计日志；
禁止 COPY PROGRAM 行为；
支持 pgvector 插件；

支持 tencentdb_superuser 插件

V14.2_r1.9

修复创建插件时提权漏洞：创建插件时 tencentdb_superuser 不再提权成 superuser

V14.2_r1.7

增加 GUC 控制用户 COPY 行为，禁止创建插件时使用 COPY 行为。

V14.2_r1.4

支持 failover slot ；

tencentdb_superuser 支持创建/修改 event_trigger ；

逻辑复制解析 drop schema 失败时抛出表 oid 信息 ；

将 session_replication_role 参数级别改为 PGC_USERSET ；

修复 dblink 同实例跨库访问权限问题。

V14.2_r1.3

支持 tencentdb_pwdcheck 强密码认证插件。

V14.2_r1.2

兼容社区14.2版本。

支持 dblink 插件。

v14.2_r1.1

兼容社区14.2版本。

支持 cos_fdw 插件。

v14.2_r1.0

兼容社区14.2版本。

合入 TencentDB PostgreSQL 内核改动能力。

支持 pg_cron 插件。

支持 TopN 插件。

支持 TDE 数据透明加密。

PostgreSQL 14 babelfish 版

v14.3_r1.19

社区 BUG 修复：备机 promote 后时间线回切导致数据损坏。

v14.3_r1.12

修复 CVE-2023-39417 安全漏洞。

v14.3_r1.10

修复创建插件时提权漏洞：创建插件时 `tencentdb_superuser` 不再提权成 `superuser`。

v14.3_r1.8

增加 GUC 控制用户 COPY 行为，禁止创建插件时使用 COPY 行为。

v14.3_r1.6

支持将 NUL 转义成自定义字符。

v14.3_r1.3

支持 `pg_basebackup` 双流合并和过滤文件；

支持 `tencentdb_superuser`；

支持 `soft_limit_connection` 软连接限制；

支持 `qps` 统计插件；

禁止 `superuser` 执行 `owner` 为普通 `user` 的函数；

支持三方插件；

禁用 `dblink`；

修复备机异步 `invalid buffer` 的 bug；

支持 `topn` 和 `pg_cron` 插件；

使用 `tencentdb_superuser` `CREATE/ALTER superuser` 时，会默认 `CREATE/ALTER` 一个 `tencentdb_superuser`；

支持 TDE；

支持服务端 `basebackup` 时过滤文件（夹）；

支持 `cos_fdw` 冷热存储分离插件；

修复 `pg_cron` 安全问题；

支持 `dblink` 同实例跨库访问；

允许 `superuser` 执行所有 `oid<16384` 的函数；

修复社区 `CIC/RIC` 索引数据丢失的问题；

支持 `babelfish` 初始内核及 2.1 版本插件代码。

PostgreSQL 13

V13.14_r1.17

`cos_fdw` 增加安全鉴权功能，限制访问同账户购买的 COS。

V13.14_r1.16

修复审计极速版影响行数一直显示为 0 的问题；
支持 sequential_uuids 插件；
修复审计精细版扩展协议下 SELECT 语句无法审计的问题；
支持 starrocks_fdw 插件。

V13.14_r1.15

兼容社区版本13.14；

V13.3_r1.14

修复审计日志长连接导致的内存泄露；
修复 dblink 非法参数导致 core；
出于安全考虑，普通日志记录 COPY 语句；
tencentdb_superuser 支持创建 replication origin 权限；
修复 cos_fdw 进行大分区表扫描时的超时报错问题。
社区 BUG 修复：备机 promote 后时间线回切导致数据损坏。

V13.3_r1.13

修复 CVE-2023-5869。

V13.3_r1.12

审计日志性能优化。

V13.3_r1.11

支持大版本升级；
更新部分插件版本。

V13.3_r1.10

兼容社区13.3版本。
修复 CVE-2023-39417 安全漏洞。
支持 tencentdb_system_stat 进程监控插件。
支持 pg_squeeze、pg_similarity 插件。

V13.3_r1.9

兼容社区13.3版本。
支持 dblink 跨实例访问。
rds_server_handler 插件支持使用命令清空所有 cache。
支持数据库全审计。

禁止 COPY PROGRAM 行为。

支持 tencentdb_superuser 插件。

V13.3_r1.8

兼容社区13.3版本。

对部分插件在创建和升级时可能造成的安全问题进行加固。

V13.3_r1.6

兼容社区13.3版本。

新增 tencentdb_failover_slot 插件，支持 failover slot 功能。

新增 tencentdb_superuser 权限，允许使用 checkpoint 命令。

新增 tencentdb_superuser 权限，允许创建 event_trigger。

支持使用 dblink 进行同实例跨库访问。

优化 walsender 解析 drop publication 语句性能。

V13.3_r1.5

兼容社区13.3版本。

支持 tencentdb_pwdcheck 强密码认证插件。

V13.3_r1.4

兼容社区13.3版本。

支持 db_link 插件。

v13.3_r1.3

兼容社区13.3版本。

支持 cos_fdw 插件。

v13.3_r1.2

兼容社区13.3版本。

支持 pg_cron 插件。

支持 TopN 插件。

支持 TDE 数据透明加密。

v13.3_r1.1

兼容社区13.3版本。

修复了一些插件管理方面的 BUG。

v13.3_r1.0

兼容社区13.3版本。
合入社区 PostgreSQL 13.3 版本的所有功能。
新增管理员角色 pg_tencentdb_superuser。
合入腾讯云关于 PostgreSQL 历史优化功能包。

PostgreSQL 12

V12.18_r1.22

cos_fdw 增加安全鉴权功能，限制访问同账户购买的 COS。

V12.18_r1.21

修复审计精细版扩展协议下 SELECT 语句无法审计的问题。

V12.18_r1.20

修复 zhparser 权限问题；
修复审计极速版影响行数一直显示为 0 的问题；
支持 sequential_uuids 插件。

V12.18_r1.19

更新社区小版本至 12.18。

V12.7_r1.18

修复审计日志长连接导致的内存泄露；
修复 dblink 非法参数导致 core；
出于安全考虑，普通日志记录 COPY 语句；
tencentdb_superuser 支持创建 replication origin 权限；
修复 cos_fdw 进行大分区表扫描时的超时报错问题。
社区 BUG 修复：备机 promote 后时间线回切导致数据损坏。

V12.7_r1.17

支持 orafce 插件。

V12.7_r1.16

修复 CVE-2023-5869。

V12.7_r1.15

审计日志性能优化。

V12.7_r1.14

支持大版本升级；
更新部分插件版本。

V12.7_r1.13

兼容社区12.7版本。
修复 CVE-2023-39417 安全漏洞。
支持 tencentdb_system_stat 进程监控插件。
支持 pg_squeeze、pg_similarity 插件。

V12.7_r1.12

兼容社区12.7版本。
支持 1.5.2 版本 pg_squeeze 插件。
升级 zhparser 插件到 2.2 版本。

V12.7_r1.10

兼容社区12.7版本。
对部分插件在创建和升级时可能造成的安全问题进行加固。

V12.7_r1.8

兼容社区12.7版本。
新增 tencentdb_failover_slot 插件，支持 failover slot 功能。
新增 tencentdb_superuser 权限，允许使用 checkpoint 命令。
新增 tencentdb_superuser 权限，允许创建 event_trigger。
支持使用 dblink 进行同实例跨库访问。
优化 walsender 解析 drop publication 语句性能。

V12.7_r1.7

兼容社区12.7版本。
支持 tencentdb_pwdcheck 强密码认证插件。

V12.7_r1.6

兼容社区12.7版本。
支持 timescaledb 插件1.7.5版本和2.6.0版本。

v12.7_r1.5

兼容社区12.7版本。
支持 dblink 插件。

v12.7_r1.3

兼容社区12.7版本。
支持 cos_fdw 插件。

v12.7_r1.2

兼容社区12.7版本。
支持 pg_cron 插件。
支持 TopN 插件。
支持 TDE 数据透明加密。

v12.7_r1.1

兼容社区12.7版本。
合入社区 PostgreSQL 12.7 版本的所有功能。

v12.4_r1.0

兼容社区12.4版本。
新增管理员角色 pg_tencentdb_superuser。
合入腾讯云关于 PostgreSQL 历史优化功能包。

PostgreSQL 11

V11.22_r1.18

新增 sequential_uuids 插件

V11.22_r1.17

兼容社区11.22版本。

V11.12_r1.16

社区 BUG 修复：备机 promote 后时间线回切导致数据损坏。

V11.12_r1.15

修复社区 CVE-2023-5869。

V11.12_r1.14

审计日志性能优化。

V11.12_r1.13

支持大版本升级；
更新部分插件版本。

V11.12_r1.12

兼容社区11.12版本。
修复 CVE-2023-39417 安全漏洞。
支持 tencentdb_system_stat 进程监控插件。
支持 pg_squeeze、pg_similarity 插件。

V11.12_r1.11

兼容社区11.12版本。
支持 dblink 跨实例访问。
rds_server_handler 插件支持使用命令清空所有 cache。
支持数据库全审计。
禁止 COPY PROGRAM 行为。
支持 tencentdb_superuser 插件。

v11.12_r1.9

兼容社区11.12版本。
对部分插件在创建和升级时可能造成的安全问题进行加固。

V11.12_r1.7

兼容社区11.12版本。
新增 tencentdb_failover_slot 插件，支持 failover slot 功能。
新增 tencentdb_superuser 权限，允许使用 checkpoint 命令。
新增 tencentdb_superuser 权限，允许创建 event_trigger。
支持使用 dblink 进行同实例跨库访问。
优化 walsender 解析 drop publication 语句性能。

V11.12_r1.6

兼容社区11.12版本。
支持 tencentdb_pwdcheck 强密码认证插件。

V11.12_r1.5

兼容社区11.12版本。
支持 dblink 插件。

v11.12_r1.3

兼容社区11.12版本。

支持 `cos_fdw` 插件。

v11.12_r1.2

兼容社区11.12版本。

支持 `pg_cron` 插件。

支持 TopN 插件。

支持 TDE 数据透明加密。

v11.12_r1.1

兼容社区11.12版本。

v11.8_r1.0

兼容社区11.8版本。

合入社区 PostgreSQL 11.8 版本的所有功能。

新增管理员角色 `pg_tencentdb_superuser`。

合入腾讯云关于 PostgreSQL 历史优化功能包。

PostgreSQL 10

V10.23_r1.17

支持 `sequential_uuids` 插件。

V10.23_r1.16

兼容社区10.23版本。

V10.17_r1.15

兼容社区10.17版本。

社区 BUG 修复：备机 `promote` 后时间线回切导致数据损坏。

V10.17_r1.14

兼容社区10.17版本。

修复社区 CVE-2023-5869。

V10.17_r1.13

兼容社区10.17版本。

大版本升级内核适配；

pgrouting 插件支持 3.2.2 版本。

V10.17_r1.12

兼容社区10.17版本。

修复 CVE-2023-39417 安全漏洞。

V10.17_r1.11

兼容社区10.17版本。

修复创建插件时提权漏洞：创建插件时 `tencentdb_superuser` 不再提权成 `superuser`。

V10.17_r1.10

兼容社区10.17版本；

增加 GUC 控制用户 COPY 行为，禁止创建插件时使用 COPY 行为。

V10.17_r1.9

兼容社区10.17版本；

支持 `dblink` 跨实例访问；

`rds_server_handler` 插件支持使用命令清空所有 `cache`。

V10.17_r1.8

兼容社区10.17版本；

支持 `failover slot`；

`tencentdb_superuser` 支持创建/修改 `event_trigger`；

针对多对象逻辑复制 `drop publication` 语句解码做性能优化；

将 `session_replication_role` 参数级别改为 `PGC_USERSSET`。

V10.17_r1.7

兼容社区10.17版本。

支持 `tencentdb_pwdcheck` 强密码认证插件。

V10.17_r1.6

兼容社区10.17版本；

修复 `tencentdb_superuser` 使用 `dblink` 的问题。

V10.17_r1.5

兼容社区10.17版本；

修复 `pg_cron` 安全问题；

支持 `dblink` 同实例跨库访问。

v10.17_r1.4

兼容社区10.17版本。
支持 cos_fdw 插件。

v10.17_r1.3

兼容社区10.17版本。
支持 pg_cron 插件。
支持 TopN 插件。
支持 TDE 数据透明加密。

v10.17_r1.2

兼容社区10.17版本。
禁用 dblink ；
修复备机异步 invalid buffer 的 bug。

v10.17_r1.1

兼容社区10.17版本。
支持 pg_basebackup 双流合并和过滤文件 ；
支持 tencentdb_superuser ；
支持备机异步 invalid buffer ；
支持 soft_limit_connection 软连接限制 ；
支持 qps 统计插件 ；
支持三方插件。

v10.4_r1.1

兼容社区10.4版本。
修复了一些插件管理方面的 BUG。

v10.4_r1.0

兼容社区10.4版本。
新增管理员角色 pg_tencentdb_superuser。
合入腾讯云关于 PostgreSQL 历史优化功能包。

查看内核版本

最近更新时间：2024-01-22 16:17:09

本文为您介绍如何查看云数据库 PostgreSQL 内核版本信息。

操作步骤

通过登录到 PostgreSQL 实例查看内核版本

1. 您可通过云服务器，连接到 PostgreSQL 实例查看内核版本，连接登录实例方式请参见 [连接 PostgreSQL 实例](#)。
2. 登录实例后执行如下命令查看内核版本。

```
show tencentdb_version;
```

注意：

部分存量实例通过命令执行失败，或者输出结果为v1。则代表此版本的 Release 版本为 r1.0，如数据库版本为 PostgreSQL 10.4，则内核版本为 v10.4_r1.0。

通过 PostgreSQL 控制台查询

1. 登录 [PostgreSQL 控制台](#)，在实例列表，单击**实例 ID**或**操作列**的**管理**进入实例管理页面。
2. 在实例详情页的**配置信息**下，可查看内核版本。

Configuration	
Architecture	Dual-Server High Availability (one-primary-one-secondary)
Database Version	PostgreSQL 12.4
Kernel Version	v12.4_r1.0
Used/Total	120.00MB/20GB
Specification	1 core 2 GiB
Billing Mode	Pay as You Go
Creation Time	2021-12-07 15:37:01
Expiration Time	--

自研内核功能介绍

逻辑复制槽故障转移（Failover Slot）

最近更新时间：2024-01-22 16:17:09

功能背景

当前，PostgreSQL 用户在使用逻辑复制的时候会创建 logical replication slot，该信息没有记录在 catalog 中，因此不会将其信息从主库同步到备库。如果发生 HA 切换，逻辑复制的 Slot 会丢失，导致逻辑复制断开，给使用带来不便。因此我们需要将逻辑复制槽进行故障转移。

功能介绍

Failover Slot 功能可以将 Slot 信息从主库同步到备库，如此可以在 HA 切换之后避免逻辑订阅断开，用户可以无感知地继续使用逻辑订阅。我们提供了一个插件 `tencent_failover_slot`，用户使用该插件可以完成 Failover Slot 的创建、转换为普通 Slot、查看 Failover Slot 信息、删除 Failover Slot 等操作。同时，我们新增了 GUC 参数方便客户针对异常的情况的设置。下面详细介绍。

创建 Failover Slot

```
create extension tencentdb_failover_slot;--首先创建插件
select pg_create_logical_gailover_slot('slotname','pluginname');--创建 Failover Slot
```

注意：

Slot 是实例级别的，而插件是 database 级别的，如果切换了 database 需要调用插件里的函数，需要重新创建 `tencentdb_failover_slot` 插件。

`pg_create_logical_failover_slot` 函数包含两个参数：

slotname：Failover Slot 的名称。

pluginname：表示 plugin 的名称。

将 Failover Slot 转换为普通 Slot

```
select transform_slot_to_nonfailover('slotname');
```

说明：

slotname：slot 名字。

需要保证该 slot 当前是 inactive 的，即没有发布订阅正在使用该 slot。

查看 Failover Slot 信息

```
postgres=# select * from pg_failover_slots;
 slot_name
-----
 fs
 fsg
(2 rows)
```

说明：

pg_failover_slots 视图包含一列，展示了当前所有的 Failover Slot 的名字，更多 Failover Slot 信息请查看 pg_replication_slots 视图。

删除 Failover Slot

```
select * from pg_drop_replication_slot('slotname');
```

说明：

slotname：slot 名称。

参数设置

新增参数 failover_slot_timeline_diverged_option，类型为 enum，方便用户针对异常情况的处理设置，设置路径为 [云数据库 PostgreSQL 控制台 > 实例详情 > 参数设置](#)。该参数默认值为 error，可选值 error、rewind。当极端情况下会发生备机日志接收速度比逻辑复制慢并发生 HA 的情况，此时如果设置 error，代表逻辑复制将暂停，发布端和订阅端都能收到报错，等待用户来处理。如果设置的为 rewind，那么逻辑复制将从切换时间点开始。

注意事项

当前只支持逻辑复制槽（logical replication slot）故障转移，不支持物理复制槽（physical slot）故障转移。

TDE 透明数据加密

最近更新时间：2024-01-22 16:17:09

功能背景

随着数据安全保护的要求越来越严格，全国各地域的信息安全保护法都提到要求数据库进行加密存储。通过数据加密防止意外数据文件遗失导致的数据泄露问题。

功能介绍

云数据库 PostgreSQL 提供透明数据加密（Transparent Data Encryption, TDE）功能，透明加密指数据的加解密操作对用户透明，支持对数据文件进行实时 I/O 加密和解密，在数据写入磁盘前进行加密，从磁盘读入内存时进行解密，可满足静态数据加密的合规性要求。加密使用的密钥由密钥管理服务 KMS 产生和管理。

[KMS](#) 是腾讯云一项保护数据及密钥安全的密钥服务，服务涉及的各个流程均采用高安全性协议通信，保证服务高安全，提供分布式集群管理和热备份，保证服务高可靠和高可用。

KMS 采用的是两层密钥体系，涉及两类密钥，即用户主密钥（CMK）与数据密钥（DATA KEY）。用户主密钥用于加密数据密钥或密码、证书、配置文件等小包数据（最多4KB）。数据密钥用于加密业务数据。海量的业务数据在存储或通信过程中使用数据密钥以对称加密的方式加密，而数据密钥又通过用户主密钥采用非对称加密方式加密保护。通过两层密钥体系，确保数据文件加密。

支持版本

内核版本 v10.17_r1.2、v11.12_r1.2、v12.7_r1.2、v13.3_r1.2、v14.2_r1.0。

适用场景

透明数据加密指数据的加解密操作对用户透明，支持对数据文件进行实时 I/O 加密和解密，在数据写入磁盘前进行加密，从磁盘读入内存时进行解密，可满足静态数据加密的合规性要求。

使用说明

开启透明数据加密功能与数据库透明加密的详细介绍，请参见 [透明数据加密](#)。

异步 invalid buffer

最近更新时间：2024-01-22 16:17:09

背景介绍

PostgreSQL 主从复制的逻辑在具有大量数据表的 DDL 变更时会导致从库恢复数据极其缓慢。为了避免此类问题，TencentDB 针对此类场景进行了优化和修改。

原理介绍

PostgreSQL 是通过物理复制实现主从复制。日志同步到备机之后，备机会解析 wal 日志，来与主库保持数据一致。在 PostgreSQL 从库在恢复一条 drop 类语句时要做的操作如下：

1. 恢复系统表，例如 pg_class, pg_attribute, pg_type 等，相当于移除表的元信息。
2. close 表对应的文件。
3. 遍历 buffer 中的页面，如果缓存的是该表的页面，则标记为 invalid，后面其他进程可以使用该页面。
4. 发异步失效消息给其他 backend，通知该表已删除。
5. 删除表对应的物理文件。

但在 PostgreSQL 内核中第3步将会执行 DropRelFileNodesAllBuffers 函数。其函数需要遍历整个 shard_buffer，以查看 buffer 是否缓存有将要删除的表的数据，将其标记为失效。而 PostgreSQL 中页面大小默认为8K，以 shard_buffer 大小16GB为例，则一共有 $16GB/8K = 200W$ 个 page，每删除一个表这里需要循环200万次，如果表上面有索引，每个索引也要循环200万次。

所以从业务上看，当存在大量表变更并且快速删除表操作的时候，由于主库可以并发执行所以感觉不出性能的影响，但是因为 PostgreSQL 的备库是单进程的 recovery，就会出现主备同步日志堆积，数据延迟的问题。

问题解决

腾讯云数据库 PostgreSQL 在 recover drop table 操作时，将表信息写入一个共享的 hash 表中。当存在表文件删除场景下，不再直接进行物理删除，将删除实际文件的动作作为异步动作存储其中。

当 invalid buffer 结束时将表从 hash 表中移除，这样，如果在此过程中发生打开文件失败，检查是否存在此 hash 表中即可。并且如果在新创建文件的时候也去遍历一下此队列，若队列中存在同名文件正在 invalid buffer，则等待即可。

而 PostgreSQL 关于表文件命名是一个 uint32 整数保存，采用的是“全局分配，局部存储”的方式，即一个实例下的所有数据库使用一个计数器生成文件号，生成的文件保存在各自库的目录下，分配时，如果当前库下已有同名文件，则尝试下一个，直到没有冲突为止，计数器绕圈后重新开始。

经过优化后，可以明显发现同类场景下主从同步性能增强了3W多倍。

进程监控

最近更新时间：2024-03-20 10:33:54

功能背景

在一些通用场景下，客户应用不仅需要数据库实例整体的监控信息，同时也需要通过实例中的进程监控信息来判断业务健康度，从而进一步做业务逻辑的决策。典型的，在 SaaS 场景中，用户的一个租户单独使用一个数据库实例的 database。业务方通过监控连接到 database 进程的资源使用情况，能进一步判断租户当前的健康度等情况。

功能介绍

云数据库 PostgreSQL 提供了数据库实例的进程监控信息。系统提供了一个视图 `tencentdb_process_system_usage`，该视图包含如下信息，说明如下：

字段名	说明
<code>pid</code>	进程号。
<code>username</code>	用户名。
<code>datname</code>	数据库名。
<code>backend_type</code>	进程类型。可能的类型为 <code>autovacuum launcher</code> , <code>autovacuum worker</code> , <code>logical replication launcher</code> , <code>logical replication worker</code> , <code>parallel worker</code> , <code>background writer</code> , <code>client backend</code> , <code>checkpointer</code> , <code>startup</code> , <code>walreceiver</code> , <code>walsender</code> and <code>walwriter</code> 。除此以外，由扩展注册的后台 Worker 可能有额外的类型。
<code>query</code>	当前正在执行的查询语句。
<code>cpu_usage</code>	当前查询采样期间的 CPU 使用率，对于采样时间间隔，系统提供参数 <code>tencentdb_system_stat.sampling_interval</code> 可以设置，该参数的单位为毫秒，默认为1000。
<code>memory_bytes</code>	当前采样期间内存使用率，单位字节。

使用说明

1. 新建插件，具体语句如下：

```
postgres => create extension tencentdb_system_stat;  
CREATE EXTENSION
```

```
postgres =>
```

2. 视图 `tencentdb_process_system_usage` 的使用跟其他视图一样，只需要登陆数据库 PostgreSQL 实例，进行查询即可，如下语句所示：

```
select * from tencentdb_process_system_usage;
```

使用 starocks_fdw 实现读写分离

最近更新时间：2024-10-28 14:56:56

云数据库 PostgreSQL 支持自研的 starocks_fdw 插件，让您能够将 starocks 中的表作为 PostgreSQL 的外部表，从而实现冷热存储分离。该插件的使用兼容 mysql_fdw 插件，以下为一个使用示例，更多详细的参数及使用请参考 [mysql_fdw 官网](#)。本文的前提是您已经部署完毕 starocks，且可以访问。同时您已经购买了云数据库 PostgreSQL 实例。

本文中，已经购买一个云数据库 PostgreSQL 实例，且在同一个 VPC 下的 CVM 上已经自建一个 starocks。下面将为您一一描述操作过程。

说明：

starocks_fdw 插件当前只支持 v13.14_r1.17 及以上的 PG13 版本。

在 starocks 上准备数据

新建表格 sr_m1：

```
mysql> CREATE TABLE sr_m1 (  
->     id int not null,  
->     str INT  
-> );  
Query OK, 0 rows affected (0.02 sec)
```

插入数据：

```
mysql> insert into sr_m1(id,str)values(1,1),(2,2),(3,3);  
Query OK, 3 rows affected (0.48 sec)  
{'label':'insert_61cc249a-758a-11ef-a99a-5254006f765d', 'status':'VISIBLE', 'txnId'
```

检查数据插入成功：

```
mysql> select * from sr_m1;  
+-----+-----+  
| id  | str |  
+-----+-----+  
|  1  |  1  |  
|  3  |  3  |  
|  2  |  2  |  
+-----+-----+  
3 rows in set (0.01 sec)
```



```
-----+-----+-----+-----+-----+-----  
16499 | 16498 | server_sr | 16385 | dbadmin | {password=123456,username=root}  
(1 row)
```

在云数据库 PostgreSQL 上创建外部表

创建表格：

```
am_sr=> CREATE TABLE sr (id INT NOT NULL, str INT) PARTITION BY RANGE (id);  
CREATE TABLE
```

创建外部表：

```
am_sr=> CREATE FOREIGN TABLE sr_m PARTITION OF sr FOR VALUES FROM (1) TO (10) SERVE  
CREATE FOREIGN TABLE
```

查询数据：

```
am_sr=> select * from sr;  
 id | str  
----+-----  
  1 |   1  
  3 |   3  
  2 |   2  
(3 rows)
```