KingbaseES 运维工具参考手册

金仓数据库管理系统 KingbaseES

文档版本：V9（V009R001C002B0014）

发布日期：2024 年 10 月 30 日

北京人大金仓信息技术股份有限公司

目　录

第 1 章　前言　1
1.1　适用读者　1
1.2　相关文档　1
1.3　术语　1
1.4　手册约定　2

第 2 章　通用环境变量参数文件　3

第 3 章　系统数据与日志收集工具　7
3.1　工具概述　7
3.2　检查项　7
3.3　适用环境　12
3.4　脚本文件说明　12
3.5　执行说明　13
3.6　信息采集打包文件　17
3.7　报告示例　17

第 4 章　问题诊断辅助工具　19
4.1　工具概述　19
4.2　脚本工具所在路径　20
4.3　配置数据库连接信息　20
4.4　工具代码及功能　20
4.4.1　chk_current_waitevent.sh　20
4.4.1.1　功能　20
4.4.1.2　显示结果样例　21
4.4.2　chk_top_waitevent.sh　21
4.4.2.1　功能　21
4.4.2.2　显示结果样例　21
4.4.3　chk_event_sql.sh　22
4.4.3.1　功能　22
4.4.3.2　显示结果样例　22
4.4.4　chk_lock.sh　22
4.4.4.1　功能　22
4.4.4.2　显示结果样例　22

4.4.5 chk_top_sql.sh . . . 23
4.4.5.1 功能 . . . 23
4.4.5.2 显示结果样例 . . . 23
4.4.6 chk_top_tableIO.sh . . . 24
4.4.6.1 功能 . . . 24
4.4.6.2 显示结果样例 . . . 24
4.4.7 chk_top_tablephysicalIO.sh . . . 25
4.4.7.1 功能 . . . 25
4.4.7.2 显示结果样例 . . . 25
4.4.8 chk_top_tableseqscanIO.sh . . . 26
4.4.8.1 功能 . . . 26
4.4.8.2 显示结果样例 . . . 26
4.4.9 chk_top_indexIO.sh . . . 26
4.4.9.1 功能 . . . 26
4.4.9.2 显示结果样例 . . . 27
4.4.10 chk_top_table_dml.sh . . . 27
4.4.10.1 功能 . . . 27
4.4.10.2 显示结果样例 . . . 28
4.4.11 chk_top_wal.sh . . . 28
4.4.11.1 功能 . . . 28
4.4.11.2 显示功能样例 . . . 29

第 5 章 辅助运维工具 30
5.1 工具概述 . . . 30
5.2 kb_modify_syscfg . . . 30
5.3 kb_backup . . . 31
5.4 logic_backup_manager . . . 33
5.4.1 脚本构成说明 . . . 33
5.4.1.1 logic_backup_manager.sh . . . 33
5.4.1.2 logic_backup_manager.conf . . . 33
5.4.1.3 .job 文件 . . . 33
5.4.2 脚本使用说明 . . . 33
5.4.3 定时任务设置 . . . 34
5.4.3.1 凝思 HA 的定时任务设置 . . . 34
5.4.4 基本原理介绍 . . . 34
5.4.5 约束 . . . 35
5.4.6 job 说明 . . . 35
5.4.6.1 专用配置参数说明 . . . 35
5.4.6.2 私有通用参数说明 . . . 35
5.4.6.3 job1 类型说明 . . . 36
5.4.6.3.1 作用 . . . 36
5.4.6.3.2 原理 . . . 36

5.4.6.3.3　专用配置参数 36
5.4.6.3.4　举例说明 37
5.4.6.3.5　示例备份存储位置 37
5.4.6.4　job2 类型说明 37
5.4.6.4.1　作用 37
5.4.6.4.2　原理 37
5.4.6.4.3　专用配置参数 37
5.4.6.4.4　举例说明 37
5.4.6.4.5　示例备份存储位置 37
5.4.6.5　job3 类型说明 38
5.4.6.5.1　作用 38
5.4.6.5.2　原理 38
5.4.6.5.3　专用配置参数 38
5.4.6.5.4　举例说明 38
5.4.6.5.5　示例备份存储位置 38
5.4.6.6　job4 类型说明 38
5.4.6.6.1　作用 38
5.4.6.6.2　原理 38
5.4.6.6.3　专用配置参数 39
5.4.6.6.4　举例说明 39
5.4.6.6.5　示例备份存储位置 39
5.4.6.7　job5 类型说明 40
5.4.6.7.1　作用 40
5.4.6.7.2　原理 40
5.4.6.7.3　专用配置参数 40
5.4.6.7.4　举例说明 40
5.4.6.7.5　示例备份存储位置 40
5.4.6.8　job6 类型说明 40
5.4.6.8.1　作用 40
5.4.6.8.2　原理 41
5.4.6.8.3　专用配置参数 41
5.4.6.8.4　举例说明 41
5.4.6.8.5　示例备份存储位置 41
5.4.6.9　还原备份功能说明 41
5.4.6.9.1　作用 41
5.4.6.9.2　使用限制 41
5.4.6.9.3　原理 41
5.4.6.9.4　专用配置参数 41
5.4.6.9.5　举例说明 42
5.4.6.9.6　示例备份存储位置 42
5.4.7　全局配置参数说明 42
5.5　kb_check 44

第 6 章　预防删除工具指南 48
6.1　问题背景 48
6.2　工具原理 48
6.3　工具构成 48
6.4　专用机约束 48
6.5　安装工具 49
6.6　设置保护 49
6.7　使用指南 49
6.8　移除工具 50
6.9　使用须知 50

版权声明 52

服务周期承诺 53

第1章　前言

KingbaseES 为方便用户使用，提供了 DBA 常见的操作脚本，用于协助 DBA 的日常运维工作。运维工具包括：系统数据与日志收集工具（kb_gathertool）、问题诊断辅助工具（kb_diagtool）、辅助运维工具（kb_modify_syscfg、kb_check、kb_backup）。

数据库软件安装完成后，工具会在 $KES_HOME/SupTools 目录下。

本文档将介绍工具的功能及使用方法。

前言部分包含以下主题：

- 适用读者
- 相关文档
- 术语
- 手册约定

1.1　适用读者

本文档主要面向使用本文档中工具的用户，包括：数据库系统管理员、开发人员。

1.2　相关文档

暂无

1.3　术语

暂无

1.4　手册约定

本文档中可能出现“注意、提示、警告、另请参阅”等标志，它们所代表的含义如下：

注意:　用于突出重要/关键信息、最佳实践等。

提示:　用于突出小窍门、捷径等。

警告:　用于传递设备或环境安全警示信息，若不避免，可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。

另请参阅:

用于突出参考、参阅等。

以下程序代码书写约定适用于本文档：

符号	说明
[]	表示包含一个或多个可选项。不需要输入中括号本身。
{}	表示包含两个以上（含两个）的候选，必须在其中选取一个。不需要输入花括号本身。
|	分割中括号或者花括号中的两个或两个以上选项。不需要输入“|”本身。
...	表示其之前的元素可以被重复。
斜体	表示占位符或者需要提供特定值的变量。
大写	表示系统提供的元素，以便与用户定义的元素相互区分。除出现在方括号中的元素外，应当按照顺序逐字输入。当然，部分元素在系统中是大小写不敏感的，因此用户可以根据系统说明以小写形式输入。
小写	表示由用户提供的元素。

第2章　通用环境变量参数文件

通用环境参数文件是指 SupTools 目录下的 suptools.ini 文件。该参数文件包含 SupTools 目录下工具的通用配置，包括：bin 目录、lib 目录、数据库连接配置等。在使用 SupTools 工具前，必须根据用户的实际环境修改 suptools.ini 文件。

参数文件适用方式：参数文件使用 INI 格式，存储工具集中使用的环境变量值。根据部署主机的安全规则，可以不存储数据库用户的口令，工具使用交互方式输入数据库用户口令。

参数文件说明：

参数组	参数名	参数说明
通用参数组 general	HOME_PATH	KingbaseES 平台的安装目录
	BIN_PATH	ksql 等工具的目录
	LD_LIBRARY_PATH	库文件目录
数据库连接组 connect	HOST	数据库主机 IP
	USER	数据库用户
	PORT	数据库端口
	DATABASE	数据库名
	PASSWORD	数据库用户密码（可空）
数据库集群组 cluster	POOL_PORT	数据库集群端口
	IP_LIST	集群的主从节点集合，以逗号分割的 IP 地址组合
数据库备份组 backup	RMAN_PATH	物理备份集的存储目录
	DUMP_PATH	逻辑备份集的存储目录
	DB_LIST	可进行逻辑备份的数据库集合，以逗号分割的数据库名组合

见续表

表 2 – 续表

参数组	参数名	参数说明
	KEEP_TIME	备份结果集的保留份数
数据库巡检组 check	CHECK_LEVEL	数据库巡检的等级，月度或季度模式（MONTH OR QUARTER）
	MODE	数据库的模式，单机或集群（singal or cluster）
	STATRESET	重置统计信息
问题诊断辅助工具 diagtool	SLEEPSECONDS	检测工作的停顿时长，默认 10 秒

参数文件 (单机用例) 样例：

```
#########################################################################
# This Script is used to check OS AND KINGBASE info;
# Before used the script ,you should set some parameters according to your env;
#########################################################################

[general]
# General parameters
# KingbaseES
HOME_PATH=/opt/Kingbase/ES/V9
# KingbaseES
BIN_PATH=/opt/Kingbase/ES/V9/Server/bin
#LIBPATH
LD_LIBRARY_PATH=/opt/Kingbase/ES/V9/Server/lib:$LD_LIBRARY_PATH

[connect]
# connection parameters
HOST=127.0.0.1
USER=system
PORT=54307
DATABASE=test

[cluster]
#DATABASE POOL_PORT
#POOL_PORT=9999
#DATABASE IPADDR LIST
#IP_LIST=192.168.20.20,192.168.20.201

[backup]
# THE PATH OF RMAN BACKUP DIRECTORY
```

```
RMAN_PATH=/home/kingbase/backup/rmanbackup
# THE PATH OF DUMP BACKUP DIRECTORY
DUMP_PATH=/home/kingbase/backup/dumpbackup
#BACKUP DATABASE LIST
DB_LIST=kingbase,bak01,db01
KEEP_TIME=7

[check]
# CHECK_LEVEL（MONTH OR QUARTER）
CHECK_LEVEL=QUARTER
# DATABASE TYPE(CLUSTER OR SINGAL)
MODE=SINGAL
# 重置统计信息
STATRESET=off

[diagtool]
#sleep times between two operation , default 10 second
SLEEPSECONDS=10
```

参数文件 (集群用例) 样例：

```
#############################################################################
# This Script is used to check OS AND KINGBASE info;
# Before used the script ,you should set some parameters according to your env;
#############################################################################

[general]
# General parameters
# KingbaseES HOME_PATH until Cluster Name
HOME_PATH=/home/kingbase/cluster/project1/cluster1
# KingbaseES
BIN_PATH=/home/kingbase/cluster/project1/cluster1/kingbase/bin
#LIBPATH
LD_LIBRARY_PATH=/home/kingbase/cluster/project1/cluster1/kingbase/lib:$LD_LIBRARY_PATH

[connect]
# connection parameters
HOST=127.0.0.1
USER=system
PORT=54321
DATABASE=test

[cluster]
#DATABASE POOL_PORT
POOL_PORT=9999
```

```
#DATABASE IPADDR LIST
IP_LIST=192.168.20.20,192.168.20.201

[backup]
# THE PATH OF RMAN BACKUP DIRECTORY
RMAN_PATH=/home/kingbase/backup/rmanbackup
# THE PATH OF DUMP BACKUP DIRECTORY
DUMP_PATH=/home/kingbase/backup/dumpbackup
#BACKUP DATABASE LIST
DB_LIST=kingbase,bak01,db01
KEEP_TIME=7

[check]
# CHECK_LEVEL（MONTH OR QUARTER）
CHECK_LEVEL=QUARTER
# DATABASE TYPE(CLUSTER OR SINGAL)
MODE=CLUSTER
# 重置统计信息
STATRESET=off

[diagtool]
#sleep times between two operation , default 10 second
SLEEPSECONDS=10
```

第3章　系统数据与日志收集工具

本部分包含以下内容：

- 工具概述
- 检查项
- 适用环境
- 脚本文件说明
- 执行说明
- 信息采集打包文件
- 报告示例

3.1　工具概述

本工具使用人员主要是数据库管理人员，旨在为数据库管理人员提供系统全面信息，实现主机侧（操作系统）与数据库侧全面化检测，并对检测结果进行自动数据收集打包。

工具存放路径：SupTools/kb_gathertool

3.2　检查项

系统数据与日志收集工具的 CheckList 项如下：

表 3: 系统数据与日志收集工具的 CheckList 项

信息大类		具体信息
操作系统信息收集	硬件配置	CPU
		memory

见续表

表 3 – 续表

信息大类		具体信息
		filesystem
		网络信息收集
		swapinfo
		其它配置
	系统参数	操作系统版本
		系统参数
		资源限制
		selinux
		logind.conf
		system.conf
		IO 调度算法
		ssh 配置
		防火墙
		多路径配置
		LVM 配置
		文件系统类型，参数
		网络配置信息
	操作系统运行信息	内存信息
		CPU 利用率
		操作系统总 IO 大小
		TOPIO
		网络性能信息
		文件系统使用率
		进程内存使用排名
		系统内存段、信号量使用

见续表

表 3 – 续表

信息大类		具体信息
		service 状态信息
		操作系统日志
		crontab
		多路径状态信息
数据库信息收集	配置信息	基本信息
		许可信息
		用户数据库
		数据库所有插件
		数据库已安装插件
		数据库参数运行情况 (非缺省值)
		数据类型字段统计
		用户对象种类统计
		用户对象占用空间的柱状图
		集群配置信息 repmgr.conf
		获取 sys_hba.conf md5 值
		获取 sys_hba.conf 配置
		获取 sys_rman 备份配置
		获取 sys_rman 备份信息
		获取 kingbase.conf md5 值
		获取 kingbase.conf 配置
		用户或数据库级别定制参数
		获取 hamgr 日志错误信息
	数据库日志	获取 kbha 日志错误信息
		当前数据库日志目录
		获取错误日志信息

见续表

表 3 – 续表

信息大类		具体信息
		获取连接请求情况
		获取认证失败情况
		慢查询统计
		慢查询分布头 10 条的执行时间
		慢查询分布尾 10 条的执行时间
		Auto_explain 分析统计
	数据库状态	输出文件系统剩余空间
		输出表空间对应目录
		输出表空间使用情况
		输出数据库空间使用情况
		TOP 10 对象使用情况
		数据库连接数统计
		总剩余连接数
		用户连接数限制
		数据库连接限制
		TOP 5 SQL：语句合计执行时间
		TOP 5 SQL：语句单次执行时间
		TOP 5 SQL：语句合计使用缓存量
		TOP 5 SQL：语句单次使用缓存量
		TOP 5 SQL：语句合计使用磁盘量
		TOP 5 SQL：语句单次使用磁盘量
		TOP 5 SQL：语句完整文本
		索引数超过 4 并且 SIZE 大于 100MB 的表
		上次巡检以来未使用或使用较少的索引

见续表

表 3 – 续表

信息大类		具体信息
		数据库统计信息, 回滚比例, 命中比例, 数据块读写时间, 死锁, 复制冲突
		检查点, bgwriter 统计信息
		数据库年龄
		表年龄
		长事务, 2PC: 活动进程
		长事务, 2PC: 预备语句
	集群运行状态	是否开启归档, 自动垃圾回收
		归档统计信息
		流复制统计信息
		流复制插槽
	安全	密码泄露检查
		检查 sys_stat_statements
		检查 sys_authid
		检查 sys_user_mappings
		检查 sys_views
		简单密码检查
		用户密码到期时间
		SQL 注入风险分析
		普通用户对象上的规则安全检查
		普通用户自定义函数安全检查
		无日志用户表
		哈希索引
		触发器
		事件触发器

见续表

表 3 – 续表

信息大类		具体信息
		检查是否使用了 A-Z a-z 0-9 _ 以外的字母作为对象名
		锁等待
		继承关系检查
	数据库报告	重置数据库的统计计数器
		重置数据库的统计信息
		获取当天数据库实例的 KWR 报告
		生成的 KWR 报告文件
		获取当前时间 10 分钟前数据库实例的 KSH 报告
		获取当前时间数据库实例的 WORKLOAD 报告

3.3　适用环境

系统环境：Linux 系统 + KingbaseES (V9)

权限要求：本系统工具需要 root 权限执行，数据库系统需要 superuser 权限执行。

执行时间：默认安全扫描大概执行时间为 2～4 分钟，程序执行时间的长度由检测文件的多少决定，有可能会存在较长的时间，请耐心等待。

兼容性：目前程序针对 Centos7.x、Kylin V10、Deepin 进行开发测试，数据库系统：KingbaseES (V9)

3.4　脚本文件说明

表 4: 脚本文件说明

文件	说明
gather.sh	主程序
gather.conf	环境变量参数文件，默认读取通用参数文件。无需修改。
gather_html.conf	HTML 显示风格参数文件，默认读取通用参数文件。无需修改。

见续表

表 4 – 续表

文件	说明
gather_linux.sh	Linux 操作系统维护工具
gather_kingbase.sh	KingbaseES 数据库系统维护工具
../suptools.ini	工具集的通用参数文件

3.5　执行说明

- 执行主程序

```
[root@localhost gather]# sh gather.sh
```

- 使用 root 用户权限，执行脚本，指定报表格式类型：

```
请指定报表格式类型
~~~~~~~~~~~~~~~~~~~~~~~~~
报告类型选择HTML还是TEXT?
输入'html'作为html格式，或'text'作为文本格式，默认'text'。
请输入报表格式：text
```

- 显示收集 Linux 系统的信息过程

```
请指定报表格式类型
~~~~~~~~~~~~~~~~~~~~~~~~~
报告类型选择HTML还是TEXT?
输入'html'作为html格式，或'text'作为文本格式，默认'text'。
请输入报表格式: text

报表格式类型 text
开始收集操作系统环境配置和运行状态等信息。

---->> CPU参数
---->> 内存参数
---->> SWAP参数
---->> 块设备，HD、LVM
---->> PCI设备
---->> 操作系统发布版本
---->> 内核版本
---->> 内核参数
---->> 用户资源限制
---->> selinux配置信息
---->> logind配置信息
---->> systemd配置信息
---->> 内核引导参数
---->> 磁盘IO调度算法
---->> SSH配置信息
---->> 防火墙服务状态信息
---->> 防火墙的zone信息
```

- 显示 Linux 操作系统，完成采集信息操作，可以看到 html 格式的采集报告，已经生成。

```
---->> logind配置信息
---->> systemd配置信息
---->> 进程的IO占用排行
---->> 进程的CPU占用排行
---->> 进程的内存占用排行
---->> 文件系统使用率
---->> 进程虚拟内存使用排名
---->> 进程固定内存使用排名
---->> 文件系统inode使用率
---->> 进程间通信使用情况(共享内存，消息队列，信号 )
---->> 进程间通信状态(共享内存，消息队列，信号 )
---->> 进程间通信资源限制(共享内存，消息队列，信号 )
---->> 服务运行状态信息
---->> 20日内操作系统日志（错误信息）
---->> cron:基于用户的作业调度器
---->> cron:基于应用的作业调度器
---->> core文件收集
---->> core文件的打包文件
---->> 多路径拓扑信息

Linux操作系统信息采集文件包 ./gather_linux.tar.gz
-rw-r--r-- 1 root root 93288 Jun 23 09:40 ./gather_linux.tar.gz

Linux操作系统巡检报告文件 /txt/suptools/kb_gathertool/./report_linux.text
-rw-r--r-- 1 root root 1176933 Jun 23 09:40 /txt/suptools/kb_gathertool/./report_linux.text
```

- 开始收集数据库系统信息时，需要指定 KingbaseES 数据库 system 用户密码。



- 显示收集数据库系统的信息过程

```
开始收集KES数据库环境配置和运行状态等信息。
;
请输入system用户密码:
---->> 基本信息
---->> 许可信息
---->> 用户数据库
---->> 数据库所有插件
---->> 数据库已安装插件
---->> 数据库参数运行情况(非缺省值)
---->> 数据类型字段统计
---->> 用户对象种类统计
---->> 用户对象占用空间的柱状图
---->> 集群配置信息
---->> 获取sys_hba.conf md5值
---->> 获取sys_hba.conf配置 /opt/Kingbase/ES/V8/data/sys_hba.conf
---->> 获取sys_rman备份配置 /home/kingbase/backup/rmanbackup/sys_backup.conf
---->> 获取sys_rman备份信息
---->> 获取kingbase.conf md5值
---->> 获取kingbase.conf配置 : /opt/Kingbase/ES/V8/data/kingbase.conf
---->> 用户或数据库级别定制参数
---->> 获取hamgr日志错误信息
---->> 获取kbha日志错误信息
---->> 当前数据库日志目录 /opt/Kingbase/ES/V8/data/sys_log
---->> 获取错误日志信息
---->> 获取连接请求情况
```

- 显示 KingbaseES 数据库系统的完成采集

```
---->> 检查 sys_views
---->> 简单密码检查
---->> 用户密码到期时间
---->> SQL注入风险分析
---->> 普通用户对象上的规则安全检查
---->> 普通用户自定义函数安全检查
---->> 无日志用户表
---->> 哈希索引
---->> 触发器
---->> 事件触发器
---->> 检查是否使用了A-Z a-z 0-9 _ 以外的字母作为对象名
---->> 锁等待
---->> 继承关系检查
---->> 获取当天数据库实例的KWR报告
---->> 生成的KWR报告文件
---->> 获取当前时间10分钟前数据库实例的KSH报告
---->> 获取当前时间数据库实例的WORKLOAD报告

KingbaseES V8 数据库系统信息采集文件包 ./gather_kingbase.tar.gz
-rw-r--r-- 1 root root 14038 Jun 23 09:42 ./gather_kingbase.tar.gz

KingbaseES V8 数据库系统巡检报告文件 ./report_kingbase.text
-rw-r--r-- 1 root root 88233 Jun 23 09:42 ./report_kingbase.text


[root@2c2f28584c30 kb_gathertool]#
```

该工具会生成以下几个文件：

- gather_kingbase.tar.gz 数据库收集信息的打包文件
- gather_linux.tar.gz 操作系统收集信息的打包文件
- report_kingbase.html 工具收集到的数据库信息的 html 格式报告
- repport_linux.html 工具收集到的操作系统信息的 html 格式报告

- 执行工具时，通过设置报告格式和用户密码，无需通过交互方式，设置变量值。

```
# sh gather.sh --format=text|html --password=<yourPassWord>
```

3.6　信息采集打包文件

自动创建存放系统信息采集文件的目录，目录名是工具执行的日期 _ 时间。

```
[root@2c2f28584c30 kb_gathertool]# ls -l
total 140
drwxr-xr-x 2 root root  4096 Jun 23 09:36 2022-06-23_0936
drwxr-xr-x 2 root root  4096 Jun 23 09:42 2022-06-23_0942
-rw-r--r-- 1 1002 1002  1273 Oct 19  2021 gather.conf
-rw-r--r-- 1 1002 1002  3364 Jun 23 09:42 gather.sh
-rw-r--r-- 1 1002 1002  3816 Oct 19  2021 gather_html.conf
-rw-r--r-- 1 1002 1002 91467 Jun 23 09:35 gather_kingbase.sh
-rw-r--r-- 1 1002 1002 26896 Jun 23 09:09 gather_linux.sh
[root@2c2f28584c30 kb_gathertool]#
```

3.7　报告示例

下面的例子，是 Linux 操作系统的 HTML 格式报告，

====>> Kingbase ES Configuration & Status & Statistics Report of Primary Node 2021-10-15 15:13 <<====
==========

~~~~>> 数据库配置信息 <<~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

----->> 基本信息

属性	属性值
时间	2021-10-15 15:13:40+08
版本	KingbaseES on x86_64-unknown-linux-gnu
版本	compiled by gcc (GCC) 4.1.2 20080704 (Red Hat 4.1.2-46)
版本	64-bit
建立版本	c5c0dcc
角色	主节点

----->> 许可信息

属性	状态	属性值
License序列号	启用	579ECD2A-B219-11EA-B860-000C2937A8A8
生产日期	启用	2020-06-19
产品名称	启用	KingbaseES V8
细分版本模板名	启用	TEST-V8.0-企业版
产品版本号	启用	V008R006C
浮动基准日期	启用	
有效期间	启用	30
用户名称	启用	北京人大金仓信息技术股份有限公司
项目名称	启用	研发内部测试
CPU检查	启用	0
MAC地址	启用	00:00:00:00:00:00
最大连接数	启用	0
并行查询	启用	0
异构数据源	启用	0
分区表	启用	0
全文检索	启用	0
审计	启用	0
透明加密	启用	0
强制访问控制	启用	0
三权分立	启用	0

第4章　问题诊断辅助工具

本部分包含以下内容：

- 工具概述
- 脚本工具所在路径
- 配置数据库连接信息
- 工具代码及功能

4.1　工具概述

问题诊断辅助工具是针对 KingbaseES 数据库开发的一套问题诊断工具。方便用户在系统故障情况下，及时了解系统特定信息。

为了保证脚本的通用性，本工具使用 bash 语法编写脚本，实际运行环境必须支持 bash。

本工具的使用需要获取查询语句及等待事件，要求开启 track_activities 参数。部分 shell 脚本还需要依赖于 extension 的开启，如：sys_stat_statement，sys_kwr，sys_ksh。以上插件的开启方法可以参照《KingbaseES 数据库性能调优指南》文档。

以下表格列出脚本及扩展组件的依赖关系。

脚本名称	扩展组件依赖
chk_event_sql.sh	sys_kwr 及相关参数 sys_stat_statements 及相关参数
chk_top_sql.sh	sys_stat_statements 及相关参数
chk_top_waitevent.sh	sys_stat_statements 及相关参数
chk_top_wal.sh	sys_stat_statements 及相关参数

如果当前数据库系统是高可用集群系统，需要将数据库集群的密码文件，复制到 root 用户的根目录下。如果数据库集群的系统用户是 kingbase 用户，则通常的复制命令如下：

```
kingbase$ su - root -c "cp $HOME/.encpwd ~ "
```

4.2　脚本工具所在路径

本工具所涉及的脚本统一放在软件安装目录 SupTools/kb_diagtool。

4.3　配置数据库连接信息

在使用辅助工具前，先配置连接信息。用户读取通用参数文件 (../suptools.ini)，设置环境变量。根据用户的实际需求，修改通用参数文件。(详见通用环境变量参数文件)

4.4　工具代码及功能

4.4.1　chk_current_waitevent.sh

4.4.1.1　功能

查看数据库当前会话的等待事件信息。通过统计各类事件数量、当前等待会话正在运行的 SQL，可以大致确定系统状态。

4.4.1.2　显示结果样例

```
============================== Session Wait Start: 2022-06-23 09:46:42.379376+08==============================
wait_event_type      wait_event                    pid        query_start
usename              application_name     client_hostname      datname              state
query
-------------------- ------------------------------ ---------- ----------------------------------------
-------------------- -------------------- -------------------- -------------------- --------------------
--------------------------------------------------------------------------------------------------------
Extension            Extension                     71
system               sys_ksh collector                                               idle


*************** Total 1 of wait event Extension ***************
############### Total 1 of wait event type Extension ###############
============================== Session Wait Done: 2022-06-23 09:46:42.383020+08==============================


============================== Session Wait Start: 2022-06-23 09:46:52.407214+08==============================
wait_event_type      wait_event                    pid        query_start
usename              application_name     client_hostname      datname              state
query
-------------------- ------------------------------ ---------- ----------------------------------------
-------------------- -------------------- -------------------- -------------------- --------------------
--------------------------------------------------------------------------------------------------------
Extension            Extension                     71
system               sys_ksh collector                                               idle


*************** Total 1 of wait event Extension ***************
############### Total 1 of wait event type Extension ###############
============================== Session Wait Done: 2022-06-23 09:46:52.410878+08==============================
```

注意:　该工具信息取自 sys_stat_activity，具体字段的含义可以参考 sys_stat_activity 定义。

4.4.2　chk_top_waitevent.sh

4.4.2.1　功能

查看过去 60 秒内的数据库等待事件，按等待事件数量排序，数据取自 perf.session_history 。

4.4.2.2　显示结果样例

```
[ [redacted]@dbhost01 kb_diagtool]$ ./chk_top_waitevent.sh

 TOP wait_event in last 60 seconds
 ============================================================
 event                                      event_count
 ------------------------------------------ ---------------
 CPU or Wait On CPU                         8
```

注意:　该工具的信息取自 perf.session_history，如果报该表不存在，表示 sys_ksh 没有正确启用。

4.4.3　chk_event_sql.sh

4.4.3.1　功能

查看过去 60 秒内与特定等待事件相关的 SQL。脚本运行时，需要特定的等待事件作为参数传入。等待事件可以根据 chk_top_waitevent.sh 结果，针对性进行分析。该脚本显示的数据取自 perf.session_history 。

4.4.3.2　显示结果样例

```
[          @dbhost01 kb_diagtool]$ ./chk_event_sql.sh 'CPU or Wait On CPU'

TOP SQL waiting for 'CPU or Wait On CPU' in last 60 seconds
================================================================================================================================================
backend_type         wait_cnt        query
-------------------- --------------- ------------------------------------------------------------------------------------------------------------
client backend       8
```

注意:　该脚本传入参数是一个等待事件，通常是chk_top_waitevent.sh 返回的 top waitevent，用于进一步查询具体哪些 SQL 导致了该等待事件。

4.4.4　chk_lock.sh

4.4.4.1　功能

检查当前数据库锁等待的会话信息，及当前会话执行的 SQL。通过层级显示，可以方便定位源头操作。

4.4.4.2　显示结果样例

```
[          dbhost01 kb_diagtool]$ ./chk_lock.sh
 +425715    transactionid  ExclusiveLock       null                          delete from t1;
 +   424182    transactionid  ShareLock           null                             delete from t1;
```

输出信息解释如下：

- 425715：表示进程 PID。

- transactioid：表示锁的类型，这里是事务锁。
- Exclusive：表示锁的模式，这里是独占锁。
- Null：表对象的名称。由于是事务锁，因此，这里为 null。
- Delete from t1 ：表示引起锁的具体 SQL。

4.4.5　chk_top_sql.sh

4.4.5.1　功能

查询过去 10 秒，最耗时、逻辑 IO 最多、物理 IO 最多（扣除 buffer hit）、临时块写入最多的 SQL，每类选取 5 条（默认 5 条），分别列出，数据取自 sys_stat_statement 。可以通过 chk_top_sql.sh $n , 返回 n 条记录。

4.4.5.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_sql.sh

TOP 5 totaltime SQL
================================================================================================================================================
totaltime       queryid                 query
--------------- ----------------------- ------------------------------------------------------------------------------------------------------
8               -9150234145477264 26    create table public.tmp_sql_stats_11223344 as select * from sys_stat_statements
4               -106690455630532856     DELETE FROM perf.ksh_history WHERE ts < (CURRENT_TIMESTAMP - $1::INTERVAL)
4               -1857704280682153859    INSERT INTO perf.ksh_statements (query_id, query) SELECT queryId, query from sys_st
2               -3370240319250023549    select setting from pg_settings where name = $1
2               788155794125789282      select $1||$2||total_time||$3||queryid||$4||query from (

TOP 5 logicalread SQL
================================================================================================================================================
logicalread     queryid                 query
--------------- ----------------------- ------------------------------------------------------------------------------------------------------
2550            -9150234145477264 26    create table public.tmp_sql_stats_11223344 as select * from sys_stat_statements
980             -1857704280682153859    INSERT INTO perf.ksh_statements (query_id, query) SELECT queryId, query from sys_st
728             -7971174217226211882    SELECT COUNT($1) AS count FROM pg_catalog.pg_class WHERE oid = $2 and relname = $3
557             788155794125789282      select $1||$2||total_time||$3||queryid||$4||query from (
422             8729941509748168300     select $1||$2||logical_read||$3||queryid||$4||query from (

TOP 5 phyicalread SQL
================================================================================================================================================
phyicalread     queryid                 query
--------------- ----------------------- ------------------------------------------------------------------------------------------------------
0               -9111137434597618251    create temporary table tmp_locked_object_1112                                                  +
0               -8534172773558095444    drop table if exists public.tmp_sql_stats_11223344
0               -7971174217226211882    SELECT COUNT($1) AS count FROM pg_catalog.pg_class WHERE oid = $2 and relname = $3
0               -7617065252667361420    declare                                                                                       +
0               -9207253799556487171    select $1||$2||tmpfile_writen||$3||queryid||$4||query from (                                   +

TOP 5 tmpfilewriten SQL
================================================================================================================================================
tmpfilewriten   queryid                 query
--------------- ----------------------- ------------------------------------------------------------------------------------------------------
0               -9111137434597618251    create temporary table tmp_locked_object_1112                                                  +
0               -8534172773558095444    drop table if exists public.tmp_sql_stats_11223344
0               -7971174217226211882    SELECT COUNT($1) AS count FROM pg_catalog.pg_class WHERE oid = $2 and relname = $3
0               -7617065252667361420    declare                                                                                       +
0               -9207253799556487171    select $1||$2||tmpfile_writen||$3||queryid||$4||query from (                                   +
```

4.4.6　chk_top_tableIO.sh

4.4.6.1　功能

查询过去 10 秒，IO 值最大的表。分别统计表的 IO，表的索引 IO，toast 对象的 IO ，数据取自 sys_statio_all_tables 。这里的 IO 包括缓冲区读取和磁盘物理读取。

表格说明：

- Schename：模式名
- Relname：统计对象名（表、索引、toast）
- xxxIO：对象的 IO 块数量

4.4.6.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_tableIO.sh
NOTICE:  table "tmp_io_stats_11223344" does not exist, skipping

 TOP 5 tableonlyIO SQL
 ================================================================================================================================
 schemaname                     relname                                                                  tableonlyIO
 ------------------------------ ------------------------------------------------------------------------ -----------------------
 perf                           ksh_statements                                                           552
 pg_catalog                     pg_class                                                                 190
 pg_catalog                     pg_attribute                                                             148
 pg_catalog                     pg_operator                                                              64
 pg_catalog                     pg_type                                                                  56

 TOP 5 indexonlyIO SQL
 ================================================================================================================================
 schemaname                     relname                                                                  indexonlyIO
 ------------------------------ ------------------------------------------------------------------------ -----------------------
 perf                           ksh_statements                                                           552
 pg_catalog                     pg_attribute                                                             310
 pg_catalog                     pg_class                                                                 145
 pg_catalog                     pg_type                                                                  142
 pg_catalog                     pg_proc                                                                  102

 TOP 5 toastonlyIO SQL
 ================================================================================================================================
 schemaname                     relname                                                                  toastonlyIO
 ------------------------------ ------------------------------------------------------------------------ -----------------------
 pg_catalog                     pg_rewrite                                                               3
 pg_catalog                     pg_description                                                           0
 information_schema             sql_sizing_profiles                                                      0
 perf                           kwr_snap_server_info                                                     0
 pg_catalog                     pg_statistic                                                             0

 TOP 5 tableallIO SQL
 ================================================================================================================================
 schemaname                     relname                                                                  tableallIO
 ------------------------------ ------------------------------------------------------------------------ -----------------------
 perf                           ksh_statements                                                           1104
 pg_catalog                     pg_attribute                                                             458
 pg_catalog                     pg_class                                                                 335
 pg_catalog                     pg_type                                                                  198
 pg_catalog                     pg_proc                                                                  151
```

4.4.7　chk_top_tablephysicalIO.sh

4.4.7.1　功能

查询过去 10 秒，物理 IO 值最大的表（数量可传参数）。分别统计表的 IO，表的索引 IO，toast 对象的 IO，数据取自 sys_statio_all_tables，与 chk_top_table IO 不同，这里的 IO 只包括磁盘物理读取。

表格说明：

- Schename：模式名
- Relname：统计对象名（表、索引、toast、表的合计）
- xxxIO：对象的 IO 块数量

4.4.7.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_tablephysicalIO.sh

 TOP 5 tableonlyphysicalIO SQL
 ================================================================================================================================
 schemaname                     relname                                                          tableonlyphysicalIO
 ------------------------------ ---------------------------------------------------------------- --------------------
 public                         tmp_io_stats_11223344                                            6
 pg_catalog                     pg_attribute                                                     1
 pg_catalog                     sys_sysprivilege                                                 0
 pg_catalog                     sys_privilege                                                    0
 perf                           kwr_snap_sql_stmt_total                                          0

 TOP 5 indexonlyphysicalIO SQL
 ================================================================================================================================
 schemaname                     relname                                                          indexonlyphysicalIO
 ------------------------------ ---------------------------------------------------------------- --------------------
 pg_catalog                     pg_inherits                                                      0
 pg_catalog                     sys_sysprivilege                                                 0
 perf                           kwr_snap_sql_stmt_total                                          0
 pg_toast                       pg_toast_1262                                                    0
 pg_catalog                     pg_amproc                                                        0

 TOP 5 toastonlyphysicalIO SQL
 ================================================================================================================================
 schemaname                     relname                                                          toastonlyphysicalIO
 ------------------------------ ---------------------------------------------------------------- --------------------
 pg_catalog                     pg_rewrite                                                       0
 pg_catalog                     pg_description                                                   0
 information_schema             sql_sizing_profiles                                              0
 perf                           kwr_snap_server_info                                             0
 pg_catalog                     pg_statistic                                                     0

 TOP 5 tableallphysicalIO SQL
 ================================================================================================================================
 schemaname                     relname                                                          tableallphysicalIO
 ------------------------------ ---------------------------------------------------------------- --------------------
 public                         tmp_io_stats_11223344                                            6
 pg_catalog                     pg_attribute                                                     1
 pg_catalog                     sys_sysprivilege                                                 0
 pg_catalog                     sys_privilege                                                    0
 perf                           kwr_snap_sql_stmt_total                                          0
```

4.4.8　chk_top_tableseqscanIO.sh

4.4.8.1　功能

查询过去 10 秒，全表扫描的 IO 数量最大的表。全表扫描的 IO 数量是 seq_scan 次数与表大小的乘积，数据取自 sys_stat_sys_tables。

表格说明：

- Schename：模式名
- Relname：统计对象名（表）
- Seq_scan：扫描对象次数
- xxxIO：对象的 IO 块数量

4.4.8.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_tableseqscanIO.sh

 TOP 20 tableseqscanIO SQL
 ================================================================================================================================
 schemaname                     relname                                  seq_scan                                 tableseqscanIO
 ------------------------------ ---------------------------------------- ---------------------------------------- ----------------
 pg_catalog                     pg_attribute                             0                                        0
 pg_catalog                     pg_proc                                  0                                        0
 pg_catalog                     pg_class                                 1                                        0
 pg_catalog                     pg_authid                                1                                        0
 pg_catalog                     pg_database                              0                                        0
 pg_catalog                     pg_am                                    3                                        0
 pg_catalog                     pg_amproc                                0                                        0
 pg_catalog                     pg_constraint                            0                                        0
 pg_catalog                     pg_index                                 0                                        0
 pg_catalog                     pg_namespace                             1                                        0
 pg_catalog                     pg_opclass                               0                                        0
 pg_catalog                     pg_trigger                               0                                        0
 pg_catalog                     pg_db_role_setting                       0                                        0
 pg_catalog                     pg_extension                             0                                        0
 pg_catalog                     pg_subscription                          0                                        0
 pg_catalog                     pg_subscription_rel                      1                                        0
 pg_catalog                     pg_publication                           0                                        0
 pg_catalog                     sys_recyclebin                           1                                        0
 pg_catalog                     pg_context                               0                                        0
 pg_catalog                     pg_tablespace                            0                                        0
```

4.4.9　chk_top_indexIO.sh

4.4.9.1　功能

与 chk_top_tableIO 不同，这里只关注哪些索引引发的 IO 最大，分别统计逻辑 IO 与物理 IO。

表格说明：

- Schename：模式名
- Relname：统计对象名（表）

- Relname：索引名
- indexIO：索引的 IO 块数量

4.4.9.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_indexIO.sh

TOP 5 indexIO SQL
================================================================================================================================
schemaname                     relname                          indexrelname                          indexIO
------------------------------ -------------------------------- ------------------------------------- --------------------
perf                           ksh_history                      ksh_history_ts_index                  1109
perf                           ksh_history                      ksh_history_queryid_index             1103
perf                           ksh_statements                   pk_ksh_statements                     661
pg_catalog                     pg_class                         pg_class_oid_index                    660
pg_catalog                     pg_attribute                     pg_attribute_relid_attnum_index       254

TOP 5 indexphysicalIO SQL
================================================================================================================================
schemaname                     relname                          indexrelname                          indexphysicalIO
------------------------------ -------------------------------- ------------------------------------- --------------------
perf                           ksh_history                      ksh_history_ts_index                  3
perf                           ksh_history                      ksh_history_queryid_index             1
pg_catalog                     pg_attribute                     pg_attribute_relid_attnam_index       1
pg_toast                       pg_toast_3456                    pg_toast_3456_index                   0
pg_toast                       pg_toast_826                     pg_toast_826_index                    0
```

4.4.10　chk_top_table_dml.sh

4.4.10.1　功能

查询过去 10 秒（时间参数，可填），增删改次数最多的表。数据取自 sys_stat_user_tables。

表格说明：

- Schename：模式名
- Relname：统计对象名（表）
- insertopeation：增删改次数

4.4.10.2　显示结果样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_table_dml.sh

 TOP 5 insertopeation SQL
 ================================================================================================================================
 schemaname                     relname                                                                  insertopeation
 ------------------------------ ------------------------------------------------------------------------ --------------
 public                         tmp_io_stats_11223344                                                    217
 pg_catalog                     pg_attribute                                                             28
 pg_catalog                     pg_depend                                                                3
 pg_catalog                     pg_type                                                                  2
 pg_catalog                     pg_class                                                                 1


 TOP 5 deleteoperation SQL
 ================================================================================================================================
 schemaname                     relname                                                                  deleteoperation
 ------------------------------ ------------------------------------------------------------------------ --------------
 pg_catalog                     pg_attribute                                                             13
 pg_catalog                     pg_depend                                                                3
 pg_catalog                     pg_type                                                                  2
 pg_catalog                     pg_class                                                                 1
 pg_toast                       pg_toast_29597                                                           0


 TOP 5 updateoperation SQL
 ================================================================================================================================
 schemaname                     relname                                                                  updateoperation
 ------------------------------ ------------------------------------------------------------------------ --------------
 pg_catalog                     pg_attribute                                                             22
 pg_toast                       pg_toast_29597                                                           0
 pg_catalog                     pg_namespace                                                             0
 pg_toast                       pg_toast_2619                                                            0
 pg_toast                       pg_toast_29995                                                           0


 TOP 5 tableallDML SQL
 ================================================================================================================================
 schemaname                     relname                                                                  tableallDML
 ------------------------------ ------------------------------------------------------------------------ --------------
 public                         tmp_io_stats_11223344                                                    217
 pg_catalog                     pg_attribute                                                             63
 pg_catalog                     pg_depend                                                                6
 pg_catalog                     pg_type                                                                  4
 pg_catalog                     pg_class                                                                 2
```

4.4.11　chk_top_wal.sh

4.4.11.1　功能

检查当前哪些 SQL 引发最多的 WALWrite 或 WALSync 等待事件，通过观察当前的等待事件，可以间接确认哪些 SQL 引发频繁 redo 日志写或 redo 日志刷盘。

表格说明：

- calls：查询次数
- times：等待事件次数
- query：查询语句

4.4.11.2　显示功能样例

```
[v8r6c5b41@dbhost01 kb_diagtool]$ ./chk_top_wal.sh

 TOP 5 WALWritecount SQL
 ================================================================================================================================
 calls           times           query
 --------------- --------------- --------------------------------------------------------------------------------------------

 TOP 5 WALSynccount SQL
 ================================================================================================================================
 calls           times           query
 --------------- --------------- --------------------------------------------------------------------------------------------
```

第5章　辅助运维工具

本部分包含以下内容：

- 工具概述
- kb_modify_syscfg
- kb_backup
- logic_backup_manager
- kb_check

5.1　工具概述

辅助运维工具提供 3 个子工具，分别实现数据库系统安装参数一键修改（kb_modify_syscfg）、数据库逻辑备份一键部署（kb_backup）、数据库逻辑备份清理脚本（logic_backup_manager）、系统巡检并自动生成 html 结果（kb_check）。

5.2　kb_modify_syscfg

实现数据库系统安装参数一键修改。kb_modify_syscfg 工具用于数据库软件安装时修改操作系统配置信息，需要用 root 用户运行。包含修改的内容如下：

实现功能	具体功能描述
修改操作系统参数	涉及内存、信号量、网络、磁盘调度算法。
Limit 限制	主要是：limits.conf , limits.d/90-nproc.conf
Selinux	修改：/etc/selinux/config
关闭防火墙	包括 firewall , iptables 服务

见续表

表 5 – 续表

实现功能	具体功能描述
创建操作系统用户	创建 kingbase 用户
数据库内存参数	配置适当的数据缓存值
检查点参数	配置与检查点相关的参数
并行参数	配置数据库语句并行执行的参数

工具存放路径：数据库软件安装目录 SupTools/kb_modify_syscfg

工具使用：进入 kb_modify_syscfg 目录，运行以下脚本。

一键修改操作系统配置，运行：./optimize_system_conf.sh

一键修改数据库配置，运行：./optimize_database_conf.sh

5.3　kb_backup

功能：实现数据库逻辑备份、部署逻辑备份定时任务工。kb_backup 工具具体实现的功能如下：

实现功能	具体功能描述
提供逻辑备份脚本	自动创建相关目录、备份数据
备份集管理	备份历史管理、备份集压缩
配置 crontab 定时任务	自动帮助配置 crontab 任务

工具存放路径：SupTools/kb_backup

工具使用：进入 kb_backup 目录。运行以下脚本：

数据库逻辑备份：运行./backup8.sh

```
[root@host202 SupTools]#
[root@host202 SupTools]# ./kb_backup/backup8.sh
kingbase database logical backup begin : 2022-09-04 12:03:27
conf file is: /home/lht/0826/kdb/KESRealPro/[redacted]2/SupTools/kb_backup/backup8.conf
test_2022090412.dmp
backup_test_2022090412.log
kingbase database logical backup end : 2022-09-04 12:03:29

[root@host202 SupTools]#
```

配置定时备份任务：运行./fast_deploy_backup8.sh

```
[root@host202 SupTools]#
[root@host202 SupTools]# ./kb_backup/fast_deploy_backup8.sh
This tool help use to deploy logical backup (backup8.sh) schedual into crontab, must execute in local server

conf file is: /home/lht/0826/kdb/KESRealPro/[redacted]/SupTools/kb_backup/backup8.conf
1.check back dest:
back dest is exist, do not create it

2.check backup8.sh script:
sucess to execute backup8.sh

3.add backup job:
do not need deploy backup8.sh in crontab
complated

4.show current crontab list:
0 2 * * * sh /l_data/lsy/iso_sit02/[redacted]2/KESRealPro/[redacted]2/SupTools/kb_backup/backup8.sh >>/l_data/lsy/i
20331SIT02/SupTools/kb_backup/logical_backup.log 2>&1 &

end
[root@host202 SupTools]#
```

检查备份情况：运行./backup8_check.sh

```
[root@host202 SupTools]#
[root@host202 SupTools]# ./kb_backup/backup8_check.sh
conf file is: /home/lht/0826/kdb/KESRealPro/[redacted]/SupTools/kb_backup/backup8.conf
This tool help us to check our logical dump file

check begin : 2022-09-04 12:06:29
check all logical dump file for every database
kdbback_dest: /home/lht/0826/kdb/backup/dumpbackup

1.start check all database's logical backup file
--------------------------------------------------------------------------
begin check database's logical backup, database name:  test
kingbase server is runing
----------------------
begin check backup file: test_2022090412.tar.gz
;
; Archive created at 2022-09-04 12:05:17 CST
;     dbname: test
;     TOC Entries: 225
;     Compression: -1
;     Dump Version: 1.14-0
;     Format: CUSTOM
;     Integer: 4 bytes
;     Offset: 8 bytes
;     Dumped from database version: 12.1
;     Dumped by sys_dump version: 12.1
;
;
; Selected TOC Entries:
;
5218; 0 0 ACL - SCHEMA sys system
5219; 0 0 ACL - SCHEMA sys_catalog system
5220; 0 0 ACL sys_catalog FUNCTION sys_config(name OUT text, setting OUT text) system
5221; 0 0 ACL sys_catalog FUNCTION sys_create_restore_point(text) system
5222; 0 0 ACL sys_catalog FUNCTION sys_current_logfile() system
5223; 0 0 ACL sys_catalog FUNCTION sys_current_logfile(text) system
5226; 0 0 ACL sys_catalog FUNCTION sys_ls_dir(text) system
5227; 0 0 ACL sys_catalog FUNCTION sys_ls_dir(text, boolean, boolean) system
5255; 0 0 ACL pg_catalog TABLE user_arguments system
5256; 0 0 ACL sys TABLE all_col_comments system
5257; 0 0 ACL sys TABLE all_col_privs system
5258; 0 0 ACL sys TABLE all_cons_columns system
5330; 0 0 ACL sys_catalog TABLE sys_enum system
5331; 0 0 ACL sys_catalog TABLE sys_event_trigger system
......
end check backup file: test_2022090412.tar.gz
----------------------

end check database test's logical backup
--------------------------------------------------------------------------

2.backup file list:
database: test, path: /home/lht/0826/kdb/backup/dumpbackup/test
-rw-r--r-- 1 root root 5.5K  9月  4 12:05 test/test_2022090412.tar.gz

2022-09-04 12:06:29 : end check all database's logical backup file

[root@host202 SupTools]#
```

5.4　logic_backup_manager

功能：实现不同类型的数据库逻辑备份、备份集清理、归档清理、基本信息收集、备份导入数据库等工作。

5.4.1　脚本构成说明

5.4.1.1　logic_backup_manager.sh

主脚本文件，解析配置文件，执行备份、清理等工作。

logic_backup_manager.sh 位于 bin 目录下。

5.4.1.2　logic_backup_manager.conf

主配置文件，全局参数：

- 常规参数，比如 ip/port/dbname/dbuser/passwd 等
- 路径参数，比如日志路径、备份路径、job 路径等
- 日志参数，比如日志级别等

logic_backup_manager.conf 位于 share 目录下。

5.4.1.3　.job 文件

每个.job 文件代表了一个备份任务，暂支持以下类型任务：

- sys_dump 备份指定模式，可多个
- sys_dump 备份指定表，可多个，可正则表达
- sys_dump 备份全库，排除指定模式、指定表
- copy 指定库的指定模式的指定表
- sys_dump 备份全库
- 备份打包指定目录下 N 天前的旧文件并删除旧文件

5.4.2　脚本使用说明

配置 `logic_backup_manager.conf` 主配置文件（优先使用和 `logic_backup_manager.sh` 同目录下的配置文件，若同目录下不存在则去加载使用 `../share/logic_backup_manager.conf` 配置文件），添加.job 备份任务文件（文件由用户创建，该文件配置参数可参考手册内相应参数）后，执行下述命令：

- logic_backup_manager.sh backup [all] # 执行所有.job 备份任务

- logic_backup_manager.sh backup myschema.job # 执行 myschema.job 备份任务
- logic_backup_manager.sh restore [all] # 执行所有备份还原任务
- logic_backup_manager.sh restore starttime-endtime # 执行指定时间备份还原任务，时间格式为 yyyymmdd
- logic_backup_manager.sh clean [all] # 清理逻辑备份内容
- logic_backup_manager.sh clean myschema.job # 清理 myschema.job 备份任务的备份内容
- logic_backup_manager.sh clean archived-wal # 清理 wal 文件
- logic_backup_manager.sh show-auto # 输出自动备份清理的 crontab 内容
- logic_backup_manager.sh backup-clean "a.job b.job" # 清理归档，执行 job 备份和清除
- logic_backup_manager.sh collect-info # 收集数据库和 OS 信息

5.4.3　定时任务设置

配置好 `logic_backup_manager.conf` 文件，设置好 job 文件，验证 job 备份任务没有问题后可使用 `show-auto` 命令输出定时任务的设置，手动添加到 crontab 列表中即可。

5.4.3.1　凝思 HA 的定时任务设置

凝思 HA 是多台机器共享一份存储的解决方案，一个时刻只能有一台机器挂载存储对外提供数据库服务，当机器 A 故障后支持自动启用另一台机器 B 对外提供服务，这时机器 B 挂载了存储。鉴于共享存储的这种特性，凝思 HA 备份清理定时任务的设置如下：

1. 将本脚本需要拷贝到共享存储上，并且配置文件也要和本脚本在同一目录下
2. 做好 `logic_backup_manager.conf` 文件配置，设置好 job 文件
3. 在挂载了共享存储的机器上执行 `show-auto` 命令，并手动设置 crontab 定时任务
4. 手动切换机器，在挂载了共享存储的新机器上执行 `show-auto` 命令，并手动设置 crontab 定时任务
5. 再次切换机器，直至在所有节点上都设置了 crontab 定时任务

注：不对外提供服务（或故障）的机器虽然也会跑定时任务，但因为缺少共享存储，是无法真正执行定时备份清理任务的

5.4.4　基本原理介绍

- 备份原理：sys_dump、copy
- 清理原理：每次执行备份任务都会在相应的 `.job.record` 文件追加本次备份内容所在的独立的目录，依据保留次数清除旧的目录

- 免密原理：ksql、sys_dump 使用链接串去连接数据库执行操作，链接串内设置了从配置文件中加载的用户名、密码、数据库名等信息
- 清理 wal 原理：SQL 查询 `archive_command` 若包含 sys_rman 则不执行清理；find 查找 `mtime+${_archived_wal_age}` 文件并排序，找到一个 `${_archived_wal_age}` 天前的最新的 wal 文件（作为锚点，该文件不会被删除），然后调用 `sys_archivecleanup` 清理 wal 归档
- 还原原理：执行 ksql 命令导入相应的 sql 文件

5.4.5　约束

脚本功能按照日期还原需要依赖文件目录中的日期（如 2022/10/10），如果修改 “logic_backup_manager.conf“ 全局配置文件中相关参数，会影响该功能的正常使用。

5.4.6　job 说明

为了满足不同客户现场的不同的逻辑备份任务，同时统一规范管理这些任务，因此将它们划分为不同的类型，现在支持 job1、job2、job3、job4、job5、job6 六种类型的任务。

每种类型的 job 有两类参数：

- 该类型 job 专用的配置参数，不可与其他类型 job 的专用配置参数重复
- 适用于所有类型 job 的私有通用参数，往往是全局配置参数的覆盖，比如 “默认保留的备份数量 `_keep_count` ”，全局配置为 30，该任务备份的数据特别重要那么可设置其私有通用参数（`_my_keep_count` ）为 365，作用域仅限当前 job

新增备份任务时，需要判断是哪种类型的 job，然后使用该类型 job 的参数做配置，并将这些内容写入到一个独立的新的.job 文件（配置参数不可重复）中。

5.4.6.1　专用配置参数说明

每种类型的 job 都有其特有的专用配置参数，不可重复，每个.job 文件必须存在某种类型的专用配置参数。

5.4.6.2　私有通用参数说明

`_my_*` 参数用于设置某个.job 文件代表的任务的私有属性，不设置则默认使用 `logic_backup_manager.conf` 内的全局配置参数。

全局配置名	私有通用参数名	说明
_keep_count	_my_keep_count	保留的备份数量，比如默认保留 30 份，某 job 任务保留 365 份
_crontab_prefix	_my_crontab_prefix	自动运行的设置，比如默认每周运行，某 job 任务每天运行
_kdb_script_log_file	_my_kdb_script_log_file	脚本日志文件设置，比如默认使用 A 文件写日志，某 job 任务使用 B 文件写日志（注：在加载主配置文件 logic_backup_manager.conf 之前警告出错信息输出到屏幕，加载主配置文件之后这些信息会写到 ${_kdb_script_log_file} 文件，加载.job 文件后若有私有的日志文件设置，后续日志会写到该私有日志文件中）
_kdb_user	_my_kdb_user	数据库用户名
_kdb_pass	_my_kdb_pass	数据库用户的密码（注：_my_kdb_pass 只有在 _my_kdb_user 存在的情况下才会生效）
_kdb_dbname	_my_kdb_dbname	数据库名
_older_files_age	_my_older_files_age	旧文件保留天数
_log_level	_my_log_level	日志级别

5.4.6.3　job1 类型说明

5.4.6.3.1　作用

备份指定模式，可使用英文逗号为间隔添加多个模式，可使用正则表达。

5.4.6.3.2　原理

1. sys_dump -n -Fc 导出指定模式的所有内容
2. sys_dump -n -Fp -s 导出指定模式的定义信息
3. tar 打包

5.4.6.3.3　专用配置参数

_dump_schemas

5.4.6.3.4　举例说明

```
# cat jobs/valid_job1.job
_dump_schemas="public,schema1"
```

5.4.6.3.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_dump/2022/07/26/valid_job1.job/
test_public_schema1_20220726_201741.dmp.tar.gz
test_public_schema1_stru_20220726_201741.sql.tar.gz
```

5.4.6.4　job2 类型说明

5.4.6.4.1　作用

备份指定表，可使用英文逗号为间隔添加多个表，可使用正则表达（注：必须是“模式名. 表”的形式）。

5.4.6.4.2　原理

1. sys_dump -t -Fc 导出指定表的所有内容
2. sys_dump -t -Fp -s 导出指定表的定义信息
3. tar 打包

5.4.6.4.3　专用配置参数

`_dump_tables`

5.4.6.4.4　举例说明

```
# cat jobs/valid_job2.job
_dump_tables="public.t1,schema1.t1"
```

5.4.6.4.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_dump/2022/07/26/valid_job2.job/
test_public.t1_schema1.t1_20220726_202143.dmp.tar.gz
test_public.t1_schema1.t1_stru_20220726_202143.sql.tar.gz
```

5.4.6.5　job3 类型说明

5.4.6.5.1　作用

备份全库，排除指定模式或指定表，可使用英文逗号为间隔添加多个表或模式，可使用正则表达（注：排除指定表参数值必须是“模式名. 表”的形式）。

5.4.6.5.2　原理

1. sys_dump -N -T -Fc 导出指定表的所有内容
2. sys_dump -N -T -Fp -s 导出指定表的定义信息
3. tar 打包

5.4.6.5.3　专用配置参数

```
_dump_all_exclude_schemas _dump_all_exclude_tables
```

5.4.6.5.4　举例说明

```
# cat jobs/valid_extbs.job
_dump_all_exclude_tables="public.t1,public.t2"
#cat jobs/valid_job3.job
_dump_all_exclude_schemas="pub*,schd*"
_dump_all_exclude_tables="hisdb.yc_*,hisdb.yc,hisdb.yc_ddd"
```

5.4.6.5.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_dump/2022/07/26/valid_job3.job/
test_pub*,schd*hisdb._20220726_202238.dmp.tar.gz
test_pub*,schd*hisdb._stru_20220726_202238.sql.tar.gz
```

5.4.6.6　job4 类型说明

5.4.6.6.1　作用

备份指定表。

5.4.6.6.2　原理

1. 若脚本所在机器与数据库服务器不在同一机器，中断该任务，因为 copy 不支持远程写文件

2. select relname from sys_class where RELNAMESPACE=(select oid from sys_namespace where nspname='$ {schema_name}') and relname like '$ {tables_prefix}' and relkind='r'　查询出要备份的表列表
3. sys_dump -t -s -Fp 备份表定义信息
4. 构造 delete 语句和用于恢复的 copy from 语句
5. 构造 copy to 语句备份每个表：copy (select * from $ {schema_name}.$ {one_table} $ {where_condition}) to ${file}

5.4.6.6.3　专用配置参数

_copy_tables_to：只能配置一个，格式为 dbname!schema!tablename，其中英文叹号! 为分隔符，tablename 可使用%，t% 表示 t 开头的该库该模式下所有的表，注意 _my_kdb_dbname 变量不会生效，以配置内的 dbname 为准；优先使用 _my_kdb_user （若未配置则使用 _kdb_user ），_my_kdb_pass （若未配置则使用 _kdb_pass ），和 dbname 构造免密字符串写到 ~/.encpwd 文件中

_copy_tables_stru：是否备份表结构，1 代表是，0 或者不配该参数则为否

_copy_tables_where_condition：备份表的部分内容

5.4.6.6.4　举例说明

```
# cat jobs/valid_copy_job4.job
_copy_tables_to="test!public!t%"
_my_keep_count="2"
# cat jobs/valid_copy_where_job4.job
_copy_tables_to="test!public!t1"
_copy_tables_where_condition="where id > 1000"

# 备份表的增量数据时，可以这样设置 .job 文件达到目的，若有多个不同的表需要增量备份，可以设置多个类似的 .job 文件：
# cat jobs/valid_time.job
_copy_tables_to="hello!public!alarm_model_%" # 备份 hello 数据库的 public 模式下的 alarm_model_% 表
_copy_tables_where_condition="where occur_time between '`date '+%Y-%m-%d' -d '-1 days'`' and '`date '+%Y-%m-%d'`'"     # 限定只备份前一天的数据
_my_crontab_prefix="30 00 * * * "      # 设置每天 0：30 执行本备份任务
_my_kdb_user="hello"                   # hello 数据库需要不同于全局配置的一个用户名去链接
_my_kdb_pass="MTIzNDU2"                # base64 后的该用户的密码
```

5.4.6.6.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_copy/2022/07/26/valid_copy_job4.job/t_20220726_202408/
copy_from  copy_to  data
# ls -A kdb_backup/databackup/kdb_daily_copy/2022/07/26/valid_copy_job4.job/t_20220726_202408/copy_from/
```

```
test_public_t1_20220726_202408_.sql  test_public_t2_20220726_202408_.sql
# ls -A kdb_backup/databackup/kdb_daily_copy/2022/07/26/valid_copy_job4.job/t_20220726_202408/copy_to/
test_public_t1_20220726_202408_.sql  test_public_t2_20220726_202408_.sql
# ls -A kdb_backup/databackup/kdb_daily_copy/2022/07/26/valid_copy_job4.job/t_20220726_202408/data/
test_public_t1_20220726_202408_.txt  test_public_t2_20220726_202408_.txt
```

5.4.6.7　job5 类型说明

5.4.6.7.1　作用

备份整个数据库。

5.4.6.7.2　原理

1. sys_dump -n -Fc 导出所有内容
2. sys_dump -n -Fp -s 导出定义信息
3. tar 打包

5.4.6.7.3　专用配置参数

`_dump_all`：注意只能设置为 yes

5.4.6.7.4　举例说明

```
# cat jobs/valid_job5.job
_dump_all="yes"
```

5.4.6.7.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_dump/2022/07/27/valid_job5.job/
test_all_20220727_203445.dmp.tar.gz  test_all_stru_20220727_203445.sql.tar.gz
```

5.4.6.8　job6 类型说明

5.4.6.8.1　作用

备份打包指定目录下 N 天前的旧文件并删除旧文件，可用于清理 sys_log 目录或者脚本日志目录。

5.4.6.8.2　原理

`find ${_tar_del_older_files_in_the_dir} -mtime +${target_older_files_age} -type f -printf '%P ' | xargs tar --remove-files -C ${_tar_del_older_files_in_the_dir} -cvzf ${tar_filename}` 找到旧文件打包，并把旧文件删除

删除天数 N 和 _older_files_age 参数有关

5.4.6.8.3　专用配置参数

`_tar_del_older_files_in_the_dir`

5.4.6.8.4　举例说明

```
# cat kdb_backup/jobs/valid_bklog1.job
_tar_del_older_files_in_the_dir="/128G/shell-script/kdb_backup/databackup/log/"
```

5.4.6.8.5　示例备份存储位置

```
# ls -A kdb_backup/databackup/older_files/valid_bklog1.job
20220805_235848.tar.gz
```

5.4.6.9　还原备份功能说明

5.4.6.9.1　作用

还原通过 job4 备份的 sql 文件，可按照时间进行选择。

5.4.6.9.2　使用限制

按时间选择是根据目录中的时间选择如 2022/10/10。如果修改配置文件中相关配置，可能导致该功能失效。

5.4.6.9.3　原理

1. 查找目录 kdb_daily_copy 下所有符合条件的 sql 文件，如果是按照时间选择还原，则通过 `echo ${file} | awk -F"_" '{print $4}` 得到文件创建日期，通过时间戳 `date -d "$var" +%s` 获得符合条件的文件
2. 通过 `ksql -f xxx.sql` 导入到数据库中

5.4.6.9.4　专用配置参数

`_kdb_daily_restore_path`

5.4.6.9.5　举例说明

```
# cat logic_backup_manager.conf
_kdb_daily_restore_path="${_kdb_backup_data_path}/kdb_daily_copy"
使用示例：
./logic_backup_manager.sh restore 20220909-20220910
```

5.4.6.9.6　示例备份存储位置

```
# ls -A kdb_backup/databackup/kdb_daily_copy
2021  2022  2023
```

5.4.7　全局配置参数说明

全局配置名	说明
_log_level	脚本日志级别：debug:0, info:1, warn:2, error:3 （debug 级别会打印 sys_dump 导出过程中
_kdb_home	KES bin/lib/share 目录的上级目录
_kdb_bin_path	sys_dump、ksql 所在的目录
_kdb_port	端口
_kdb_host	数据库的 ip 或 hostname
_kdb_connect_timeout	链接数据库超时秒数设置，必须为纯数字，0 或负数是永不超时，2 是最小超时秒数，设为
_kdb_user	数据库用户名
_kdb_pass	_kdb_user 用户名的 base64 后的密码
_kdb_dbname	数据库名
_ld_library_path	Lib 目录
_kdb_backup_jobs_path	.job 文件所在的目录
_keep_count	保留的备份数量
_kdb_os_ip_cmd	ip 命令
_os_rm_cmd	rm 命令
_os_sed_cmd	sed 命令

表 6 – 续表

全局配置名	说明
_os_base64_cmd	base64 命令
_kdb_os_grep_cmd	grep 命令
_kdb_backup_path	备份所在根目录
_kdb_backup_data_path	备份目录
_date_ymd	年月日的字符串
_date_y	年的字符串
_date_m	月的字符串
_date_d	日的字符串
_date_path	按年、月、日划分的目录的字符串
_kdb_backup_data_tmp_path	备份使用的临时目录
_kdb_backup_job_record_path	备份任务的记录文件所在的目录
_kdb_daily_dump_path	sys_dump 任务的数据目录
_kdb_daily_copy_path	copy 任务的数据目录
_kdb_backup_log_path	日志目录
_kdb_script_log_file	日志文件名
_copy_data	copy 数据的目录
_copy_to	copy to SQL 文件所在的目录
_copy_from	copy from SQL 文件所在的目录
_archived_wal_age	归档 wal 日志保留天数
_archived_wal_path	归档 wal 日志所在的目录
_crontab_prefix	crontab 自动执行的前缀
_older_files_age	旧文件保留天数
_kdb_backup_older_files_path	旧文件打包备份路径
_collect_info_path	信息收集目录

表 6 – 续表

全局配置名	说明
_collect_locks_file	locks 路径和名称
_collect_lac_file	lac 路径和名称
_collect_iostat_file	iostat 路径和名称
_collect_top_file	top 路径和名称
_collect_sys_stat_file	sys_stat 路径和名称
_collect_vmstat_file	vmstat 路径和名称
_collect_lx_file	lx 路径和名称
_collect_client_file	client 路径和名称
_collect_crontab_prefix	信息收集功能 crontab 的前缀

5.5　kb_check

功能：实现数据库和操作系统的巡检功能，并生成 html 格式报告。本工具的面向的是普通数据库用户，报告结果简明扼要，html 格式使得报告容易理解。

kb_check 工具实现如下数据项的收集：

巡检项	具体功能描述
部分数据库参数信息	包括：版本，大小写敏感，主数据库目录
操作系统运行信息	包括：CPU，磁盘，内存，IO，TOP CPU 进程，TOP memory 进程
操作系统配置信息	包括：内核参数，limits 限制，selinux，IO 调度算法，防火墙
数据库状态信息	包括：数据库大小，表空间大小，数据库活动状态，锁，license 信息
集群信息	包括：节点状态信息，流复制信息，xlog 差异检查

工具存放路径：SupTools/kb_check

工具使用：通过读取通用参数文件 (../suptools.ini)，设置环境变量。用户也可根据实际需求，修改通用参数文件。

工具执行说明：以 root 用户执行 autocheck.sh。

如果使用参数文件的数据库用户口令，则执行下面的命令：

```
#sh autocheck.sh
```

如果参数文件中，没有设置数据库用户口令，可以在命令行设置密码参数：

```
#sh autocheck.sh --password=<yourPassword>
```

如果数据库用户口令为空，或错误，会询问数据库用户口令，等待用户输入正确的口令。

工具报告成果说明：

1. 工具运行结果包括检测日志的文本文件 (check_<MONTH|QUARTER>_YYYYMMDD.log) 和检测报告的 HTML 文件（report_<MONTH|QUARTER>_YYYYMMDD.html）。
2. 日志文本文件如下：

```
|++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+|
|                                                |
|                操作系统 QUARTER 巡检信息                |
|                                                |
|++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+|

----->>>---->>>  主机名:
host202

----->>>---->>>  主机网络信息:
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        inet6 fe80::42:9eff:fe50:1abe  prefixlen 64  scopeid 0x20<link>
        ether 02:42:9e:50:1a:be  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 19  bytes 1466 (1.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eno1: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether b4:05:5d:36:ea:36  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device memory 0x80062000000-8006201ffff

eno2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.2.26  netmask 255.255.255.0  broadcast 192.168.2.255
        inet6 fe80::1e2b:801b:dee5:f8b4  prefixlen 64  scopeid 0x20<link>
        ether b4:05:5d:36:ea:37  txqueuelen 1000  (Ethernet)
        RX packets 35441934691  bytes 7609268231073 (6.9 TiB)
        RX errors 0  dropped 36938  overruns 77452201  frame 0
        TX packets 50358187151  bytes 29911803312817 (27.2 TiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device memory 0x80062020000-8006203ffff
```

3. HTML 格式报告如下：

人大金仓数据库产品运行指标分析					
数据库类型	KingbaseES V8	数据库版本号	[redacted]	数据库启动时间	2022-03-24 10:31:41.946472+08
数据库授权信息	TEST-企业版 V8R6 90 试用	数据库大小（业务库）	esrep 12 MB security 12 MB test 12 MB	数据库部署架构	单机
操作系统类型	GNU/Linux aarch64	cpu型号	型号名称： FT-2000+/64	内存大小	259673MB
操作系统版本	Kylin Linux Advanced Server release V10 (Tercel)	cpu数量	64	Swap空间	4095 MB
指标		巡检项	运行指标	阈值	健康状态
运行类指标	OS	磁盘使用情况	/ 目录已用%: 51%	<80%	正常
		inode节点	/ 目录已用%: 1%	<80%	正常
		top平均负载	Cpu(s): 0.33 % Busy	<10	正常
		防火墙状态	Active: inactive (dead)	disable	正常
		dmesg日志		err|ERROR|FAIL|fail -->waring	正常
		selinux		disable	正常
		内存使用情况	Used: 2%	<70%	正常
		swap使用情况	Used: 45%	<50%	正常
		logined配置	IPC=no	IPC=no	正常
		磁盘IO调度策略		deadline	正常
	数据库	数据库事务号	test 48	<2亿	正常
		数据库长连接（active或IDLE in transaction状态）		<两小时	正常
		数据库当前连接数	13	max*80%	正常
		数据库等待锁数量		<1	正常
		数据库日志检查		err|ERROR|FAIL|fail -->waring	正常

第6章　预防删除工具指南

6.1　问题背景

因为某些场景下，操作人员有可能误删除重要的文件或文件夹，导致整个数据库系统发生严重的问题，使用 rm-protect 这一机制，预防重要文件被误操作删除。

6.2　工具原理

使用 alias rm 的方式，操作人员命令行使用 rm 时，重定向到具有强制确认的删除文件命令。安装工具时将在系统目录生成一个配置文件

```
echo "alias rm=sys_rm" >> /etc/profile.d/rm_protect.sh
```

6.3　工具构成

工具包含三个文件，均在 KingbaseES/bin/目录下:

install-rm-protect.sh
: 安装工具到系统相关配置文件

sys_rm
: 具有强制确认的删除文件命令

sys_protect
: 设置某个文件或文件夹的工具命令

6.4　专用机约束

专用机在安装数据库管理系统时默认将完成 rm 工具安装，不需要手工再次执行安装

专用机移除 rm 工具只需要移除/etc/profile.d/rm_protect.sh 文件

基于不同操作系统，我们建议安装 RPM 之后和移除 rm 工具之后，均重启一次主机

6.5　安装工具

专用机不需要进行安装，由 RPM 自动安装此工具。需要在安装 RPM 之后重新启动主机一次，才能完成使用此预防删除工具。

必须使用 root 用户进行安装。

KingbaseES 安装完成后，需要执行一次安装命令：

```
cd <KingbaseES>/bin/
./install-rm-protect.sh
```

重新安装之前只需要执行一次，重复执行没有伤害性的效果。

删除保护将在执行此安装步骤后，重新登录系统的 session 或进程生效，强烈建议重启主机一次。

6.6　设置保护

sys_protect 针对需要保护的文件或文件夹需要单独执行此命令去保护它。设置时需要输入一个问题和回答。设置去保护某个文件夹

```
./sys_protect  /path/to/some_vip_file
Question for /path/to/some_vip_file sure to delete ?
Answer: YES
```

个别操作系统不支持相对路径设置保护，建议使用全路径文件名。

6.7　使用指南

```
rm -rf ./data/
/home/data: sure to delete ?
Answer: wrong
Wrong answer! /home/data will not be removed
The correct answer is stored in /home/.data.rm-protection

rm -rf ./data/
/home/data: sure to delete ?
Answer: right
```

6.8　移除工具

移除工具，还原 rm 命令为系统原生 rm 文件命令。

使用 root 用户：

- /bin/rm /etc/profile.d/rm_protect.sh
- 删除 /etc/bashrc 文件中 alias rm 所在的行
- 删除 /etc/bash.bashrc 文件中 alias rm 所在的行

清空此文件然后重启主机，即可卸载（或者称为暂停此功能），

本文所指的移除，是停用此功能，并不包含去删除二进制文件。已经设置删除保护的文件，不在继续具有删除保护的效果。

专用机，使用 root 用户：

```
/etc/profile.d/rm_protect.sh
```

如果专用机的权限管理禁止清空此文件，则此环境不具备卸载工具的能力。

6.9　使用须知

- 支持对文件夹的保护

  但是不自动覆盖文件夹内部的子文件夹和文件，仅在使用 `rm -rf <dir_name>` 命令时，保护才有效。使用 `rmdir` 命令删除文件时，保护无效。

- 建议卸载数据库管理软件前，先移除 rm 保护工具。

  直接卸载数据库管理软件后会导致 kingbase/bin 目录下的 sys_rm 被删除。此后再执行 rm 操作，将提示 sys_rm 不存在。

- 删除命令的原始选项导致失败

  假设存在文件夹 dir1 ，并且已通过 `sys_protect dir1` 为其设置了问答保护。执行 `rm dir1` 删除操作会触发保护问题，答对则清除该文件夹的保护标签，但不能直接删除该文件夹，需要再次执行 `rm -r dir1` 进行删除。

  > **警告**: 再次执行 `rm -r dir1` 操作时，由于 dir1 文件夹的保护标签已被清除，系统将不会进行保护校验，请谨慎操作。

- mv 操作将导致文件保护失效

  防误删保护生效判定是基于文件进行防误删保护配置时输入的全路径名，所以 mv 操作会如下效果：

  1. 文件 A 进行 mv 操作后变为文件 A1，移动后的文件 A1 不被保护。

2. 文件 A 进行 mv 操作后，再次在原路径创建同名文件 A，同名文件 A 继承原有的保护配置。

- 同时删除多个文件时的行为

  执行 `rm -rf /abc/*` 命令同时删除多个文件，如果其中部分文件未受保护（如/abc/a）、部分文件受到保护（如/abc/b），则会产生如下场景：

  – 提示是否删除 b，如果执行 Ctrl+C ，则操作取消，文件 a 和 b 都保留。

  – 提示是否删除 b，如果错误回答保护问题，则操作退出，文件 a 被删除、b 保留。

  – 提示是否删除 b，如果正确回答保护问题，则操作继续，文件 a 和 b 都被删除。

版权声明

中电科金仓（北京）科技股份有限公司（简称：电科金仓）版权所有，并保留对本手册及本声明的一切权利。

未得到电科金仓的书面许可，任何人不得以任何方式或形式对本手册内的任何部分进行复制、摘录、备份、修改、传播、翻译成其他语言、将其全部或部分用于商业用途。

免责声明

本手册内容依据现有信息制作，由于产品版本升级或其他原因，其内容有可能变更。电科金仓保留在没有任何通知或者提示的情况下对手册内容进行修改的权利。

本手册仅作为使用指导，电科金仓在编写本手册时已尽力保证其内容准确可靠，但并不确保手册内容完全没有错误或遗漏，本手册中的所有信息也不构成任何明示或暗示的担保。

技术支持

- 电科金仓官方网站：http://www.kingbase.com.cn/
- 电科金仓文档中心：http://help.kingbase.com.cn/
- 全国服务热线：400-601-1188
- 电科金仓技术支持与反馈信箱：support@kingbase.com.cn

服务周期承诺

由于市场需求在不断变化，技术创新和发展的进程不断加剧，产品的版本更迭不可避免。电科金仓对于产品版本生命周期的有效管理，有助于您提前规划项目，更好地从产品服务终止上过渡。

表 7: KingbaseES 产品生命周期里程碑

关键里程碑点	定义
产品发布日期	产品正式发布版本，即 GA（general availability）版本的发布日期。
停止销售日期	正式停止销售的日期，版本停止接受订单日。该日之后，产品将不再销售。
停止功能升级日期	在该日期之后，不再提供新特性和新硬件支持。但依旧提供错误修复、安全修复、功能维护等服务。
停止功能维护日期	在该日期之后，不再维护功能，修复问题。但依旧提供安全修复等服务
停止安全维护日期	在该日期之后，不再发布补丁版本修复中高风险漏洞，仅提供有限的支持。
产品服务终止日期	停止提供产品服务和支持的日期。包括软件维护版本，缺陷修复，以及针对该产品的所有服务支持（包括服务热线和远程/现场支持）。

服务周期策略

金仓数据库管理系统 KingbaseES 产品确保以下的服务周期：

1）产品自发布之日起至产品停止功能升级（包含新特性、新硬件支持）之日止不少于 5 年。

2）产品停止功能升级之日起至产品停止功能维护（主要包括问题修复）之日止不少于 4 年。

3）产品功能维护停止之日起至产品停止安全维护（包括中高风险漏洞修复）之日止不少于 2 年。

针对金仓数据库管理系统 KingbaseES V9 产品（发布日期：2022 年 7 月 15 日），其具体服务周期安排如下：

- 产品销售日期：以产品销售合同所约定日期为准。
- 停止升级日期：自产品销售日期起计算，满 5 年之日止。
- 服务终止日期：自产品销售日期起计算，满 11 年之日止。

服务终止策略

金仓数据库管理系统 KingbaseES 产品确保在销售后，至少提供 6 年的服务支持。

注意: 电科金仓将会综合各方因素来确定产品服务终止日期。并将在实际产品服务终止日期之前至少 90 天，通过公开方式宣布产品服务终止日期。