Kingwow（6.X）分布式数据库
系统安装手册

KINGWOW

北京自然原数科技有限公司

版权所有 © 2025 北京自然原数科技有限公司。保留一切权利。
非经北京自然原数科技有限公司书面许可，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

Kingwow、Kingwow!是北京自然原数科技有限公司的商标或者注册商标。

在本文档中以及本文档描述的产品中，出现的其他商标、产品名称、服务名称以及公司名称，由其各自的所有人拥有。

免责声明

本文档信息仅供参考，不构成任何要约或承诺。本文档可能含有预测信息，由于实践中存在很多不确定因素，可能导致实际结果与预测信息有差别。本文档的内容在未经通知的情形下可能会发生改变，敬请留意。

文档名称：	Kingwow(6.X)分布式数据库系统安装手册		
创建者：	运维交付部成员	创建日期：	
复核者：	交付运维部	复核日期：	
审批者：	交付运维部	审批日期：	

修改历史：

版本号	修改内容	撰写者	发布日期
1.0	创建	交付运维部	

KINGWOW

目录

1. 简介 1
1.1. 原理及架构 1
2. 系统规划 3
2.1. 软硬件要求 3
2.2. 部署模式 4
2.3. 安装规划 5
2.4. 服务器规划 5
2.5. 目录规划 6
2.6. 磁盘挂载点规划 6
3. 配置安装环境 7
3.1. 操作系统配置 7
3.1.1. 关闭 THP（透明大页） 7
3.1.2. 配置内核参数 9
3.1.3. 配置时间同步 9
3.1.4. 配置资源限制 12
3.2. 创建安装用户 14
3.3. 配置环境变量 15
3.4. 配置服务器间免登录 16
3.4.1. 配置过程 16
3.4.2. 验证 16
4. RPM 包安装 16
4.1. 复制安装包 17
4.2. 安装 KINGWOW 数据库 17
4.3. 创建数据目录 17
5. 启动与管理系统 18
5.1. 数据库服务启动命令 18
5.1.1. MG 启动命令 18
5.1.2. TG 启动命令 19
5.1.3. KG 启动命令 19
5.2. 单集群启动示例 20
5.3. 多集群启动示例 21
5.4. 使用 KINGWOW 客户端 23

5.5. 创建图形化客户端需要的视图 23
5.6. 配置 crontab 清理数据库垃圾文件及日志文件 23
KINGWOW

1. 简介

KINGWOW 是一个新型的分布式关系型数据库系统，支持多活部署，不仅具有良好的扩展性、高可用性、支持关系模型、支持事务处理、采用标准 SQL 接口，同时不需要用户进行分库分表，没有跨节点操作限制，因此更适合于大规模结构化数据管理和高并发事务处理的应用场景。

1.1. 原理及架构

KINGWOW 是可运行于 x86/arm 服务器集群上的分布式数据库系统，通过对传统关系数据库系统功能的拆分，实现了数据的分布式存储和查询的分布式处理，能够支持海量结构化数据的管理。如图 1 所示：

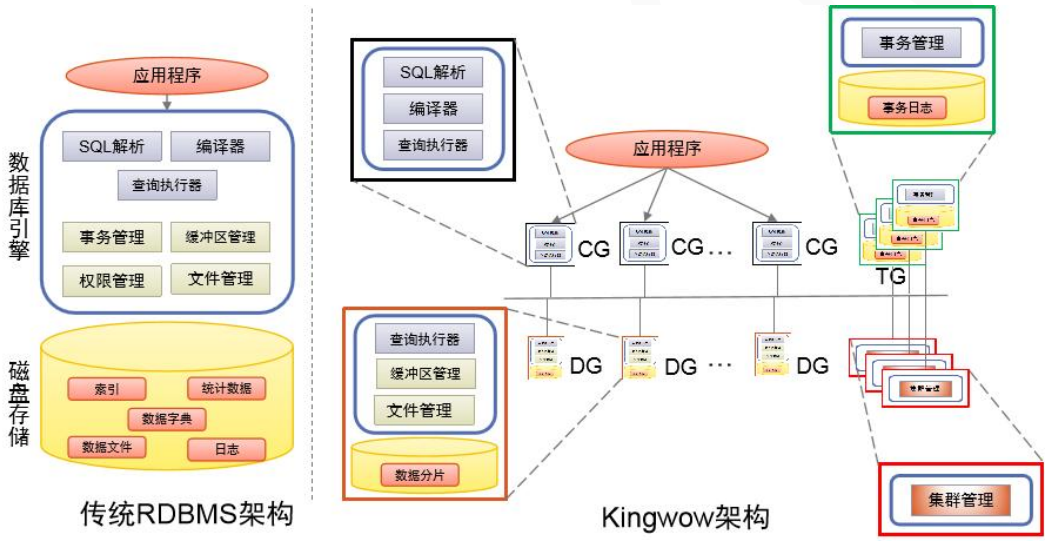


图 1 KINGWOW 与传统 RDBMS 架构对比

KINGWOW 集群中存在 3 种角色的服务进程，其中 KG（CG 与 DG 的融合）进程实现了传统数据库的查询解析、编译和执行的功能和数据文件的分布式存储与访问功能；TG 进程基于内存引擎实现了高吞吐量的事务处理功能；MG 实现了集群管理功能。集群中通常部署多个 KG 节点，通过添加节点数量可以扩展系统的存储和查询处理能力。集群中可以部署多个 TG 节点，采用基于 Quorum 的提交协议，能够保证在多数派节点正常的情况下不丢失数据。集群中同样可以部署多个 MG 节点，通过 Paxos 协议来实现自动选主，保证了集群的高可用。

ManEng：简称 MG，系统的控制节点，负责集群管理和数据分布。

集群中可以设置多个 MG 节点（一主多备），通过 Paxos 协议实现自动选主，能够保证集群的高可用；

通过租约机制（Lease）来管理集群中的 TG 节点，保证同一时刻只有一个 TG 提供事务处理服务；

通过心跳机制（Heartbeat）来管理集群中的 KG 节点，感知节点的上下线；

KINGWOW 中的数据表按主键范围被划分为多个子表（称为 Tablet），每个子表可以有多个副本，分别存储在不同的 KG 节点上。MG 通过 RootTable 结构来记录 Tablet 在 KG 上的分布位置信息，并管理 Tablet 的副本数量、复制、迁移和合并；

TransEng：简称 TG，系统中的内存事务处理引擎。

与传统数据库系统一样采用 WAL（Write-Ahead Logging）技术来保证事务的原子性和持久性；

集群中可以部署多个 TG 节点（一主多备），由主 MG 通过租约来确定主 TG，主 TG 的日志采用基于 Quorum 协议的方式同步到备 TG，能够保证在超过半数 TG 正常的情况不丢失任何数据；

事务处理产生的增量数据被保存在 TG 的内存中，通过合并操作与 KG 中保存的静态数据合并，产生新版本的静态数据；

支持跨表跨行的事务，不存在分布式事务，不需要两阶段提交协议；

采用多版本并发控制协议（MVCC）实现 Read Committed 事务隔离级别；

KwowEng（CG 与 DG 的融合）：简称 KG，实现了查询处理功能（CG）与数据的分布式存储和访问（DG）。

支持通讯协议，对 SQL 语句进行解析生成执行计划，并管理语句执行计划的运行；

缓存 Tablet 的分布位置信息，将 SQL 语句涉及的数据请求同时发送给对应的 KG，并将所有 KG 返回的结果合并；

KG 本身无状态，宕机不会对应用系统产生影响，应用可以通过其它 KG 获得服务；

存储和管理 Tablet 副本，提供本地数据查询服务，支持数据缓存；

响应 KG 的数据请求时，会从 TG 拉取对应的增量数据，融合后得到数据的最新状态，返还给 KG；

在启动数据合并操作后，KG 将会从 TG 拉取所有本地数据对应的增量数据，与本地数据合并后产生新版本的 Tablet；

数据合并操作期间，KG 正常提供数据查询服务；

2. 系统规划

2.1. 软硬件要求

虽然使用一台物理机就可以安装和运行 KINGWOW 系统，但这样的环境多用于学习和测试。为了充分发挥分布式系统的优势，建议使用 3 台以上的物理机环境。由于集群中不同类型的节点所消耗的主要系统资源不同，在规划 KINGWOW 数据库集群时，通常将 MG 和 TG 安装在同一台物理机上，将 KG 安装在同一台物理机上。

安装 KINGWOW 数据库系统所需的软硬件环境如表 1-1 所示：

表 1-1：KINGWOW 数据库系统软硬件配置需求

		MG/TG	KG
最低配置	数量	3	3
	系统	Red Hat Enterprise Linux Server release 6.8 (64bit)及以上版本[注]	
	CPU	8 核	8 核
	内存	32GB	32GB
	磁盘	SAS * 4	SAS * 4
	网络	千兆	千兆
推荐配置	数量	3	3
	系统	Red Hat Enterprise Linux Server release 6.8 (64bit)及以上版本[注]	

	CPU	32 核	32 核
	内存	128GB	128GB
	磁盘	SSD * 8	SSD * 8
	网络	万兆	万兆

注：也可使用具有相同内核的 CentOS 操作系统；

2.2. 部署模式

KINGWOW 分布式数据库系统支持多种部署方式，见表 1-2。用户可以根据需要选择合适的部署方式，并且能够在使用过程中进行灵活的调整。

表 1-2：KINGWOW 数据库系统的部署方式

部署方式	说明
单机	安装在一台服务器上，只启动 1 个 MG、1 个 TG、1 个 KG 进程。这种方式能够保证 KINGWOW 正常地运行，但无法体现出分布式系统的优势，通常只用于学习和测试的场景。
单机房（单 MG/TG）	安装在同一机房的多台服务器上，其中一台服务器上启动 MG 和 TG 进程，其它每台服务器上启动 KG 进程。这种方式无法处理 MG/TG 节点宕机的场景，只能用于对可用性要求不高的应用系统。
单机房（多 MG/TG）	安装在同一机房的多台服务器上，其中多台服务器分别启动主备 MG/TG 节点（通常为 1 主 2 备），其它每台服务器上启动 KG 进程。这种方式可以满足大多数应用系统对可靠性和可用性的需求。
多机房	安装在多个机房的多台服务器上，通常每个机房中都部署

	有 MG、TG、KG。这种方式能够满足应用系统对异地容灾的需求，当主机房发生故障时，可以自动将备机房切换为主机房，继续对外提供服务。

2.3. 安装规划

在安装 KINGWOW 数据库之前，数据库管理员首先应当对系统的资源和环境进行规划，包括服务器的使用，IP 地址和端口、安装目录、数据目录和磁盘挂载点等内容。

2.4. 服务器规划

数据库管理员可以根据示例表 1-3 对 KINGWOW 系统所使用的服务器进行规划，包括服务器所属集群，IP 地址、网卡名称、安装用户、安装目录、所运行服务、使用端口等信息。

表 1-3：KINGWOW 数据库系统服务器使用规划示例

服务器
IP 地址：192.168.1.2 网卡名称：eth0 安装用户：admin 用户密码：KINGWOW@123 安装目录：/home/admin/kingwow 应用名称：kwtest
IP 地址：192.168.1.3 网卡名称：eth0 安装用户：admin 用户密码：KINGWOW@123 安装目录：/home/admin/kingwow 应用名称：kwtest
IP 地址：192.168.1.4

网卡名称：eth0

安装用户：admin

用户密码：KINGWOW@123

安装目录：/home/admin/kingwow

应用名称：kwtest

2.5. 目录规划

数据库管理员需要规划和管理每台服务器上不同服务存储数据的目录位置，并且需要根据数据规模、使用频率、访问性能等需求将数据目录与相应的磁盘挂载点关联起来。KINGWOW 系统中的服务需要持久化到磁盘上的内容包括日志和数据，表 1-4 是一个目录规划的示例。

表 1-4：KINGWOW 数据库服务目录使用规划示例

数据库服务	类型	存储位置
MG	数据	/home/admin/kingwow/data/mg
TG	数据	/home/admin/kingwow/data/tg_data/raid0/store0 /home/admin/kingwow/data/tg_data/raid0/store1 /home/admin/kingwow/data/tg_data/raid1/store0 /home/admin/kingwow/data/tg_data/raid1/store1 … … /home/admin/kingwow/data/tg_data/raidn/store0 /home/admin/kingwow/data/tg_data/raidn/store1
	日志	/home/admin/kingwow/data/tg_commitlog
KG	数据	/home/admin/kingwow/data/dg_data/1 /home/admin/kingwow/data/dg_data/2 … … /home/admin/kingwow/data/dg_data/n

2.6. 磁盘挂载点规划

KINGWOW 系统中的 TG 服务和 KG 服务分别需要存储大规模的增量数据和静态

数据，建议不与操作系统和其它服务共享磁盘，而使用单独的磁盘来存储这些服务的数据。数据库管理员需要对每台服务器上的磁盘挂载点进行规划，并且能够与目录规划相对应。

例如，在一台运行了 KG 的服务器上，规划 8 块磁盘用于专门存储 KG 的数据，数据库管理员可以在 KINGWOW 的安装目录下创建 KG 的数据目录，如下：

- /home/admin/kingwow/data/dg_data/1
- /home/admin/kingwow/data/dg_data/2
- ……
- /home/admin/kingwow/data/dg_data/8

然后将 8 块磁盘的挂载点设为：

- /data/1
- /data/2
- ……
- /data/8
- 最后在挂载点目录与数据目录之间建立链接：
- ln -s /data/1 /home/admin/kingwow/data/dg_data/1
- ln -s /data/2 /home/admin/kingwow/data/dg_data/2
- ……
- ln -s /data/3 /home/admin/kingwow/data/dg_data/3

3. 配置安装环境

为了能够正常运行 KINGWOW 数据库系统，数据库管理员需要对安装环境进行必要的配置，包括设置网卡、内核参数、时间同步、环境变量等。

3.1. 操作系统配置

3.1.1. 关闭 THP（透明大页）

1. 检查 THP 是否开启

grep -i HugePages /proc/meminfo

cat /sys/kernel/mm/transparent_hugepage/enabled

如下图所示，AnonHugePages 为 0，enabled 为[never] 说明 THP 已关闭，否则执行 2-5 步

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# grep -i HugePages /proc/meminfo
AnonHugePages:         0 kB
HugePages_Total:       0
HugePages_Free:        0
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
[root@localhost ~]#  cat /sys/kernel/mm/transparent_hugepage/enabled
always madvise [never]
[root@localhost ~]#
```

2. vi /etc/default/grub

在 GRUB_CMDLINE_LINUX 一行最后添加 transparent_hugepage=never，如下图所示：

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb quiet transparent_hugepage=never"
GRUB_DISABLE_RECOVERY="true"
```

3. 查看系统是 BIOS 引导还是 UEFI 引导

[-d /sys/firmware/efi] && echo UEFI || echo BIOS

4. 重建 grub.cfg

如果使用的是 BIOS，执行以下命令：

grub2-mkconfig -o /boot/grub2/grub.cfg

如果使用的是 UEFI，执行以下命令，注意替换成不同操作系统

grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
或

grub2-mkconfig -o /boot/efi/EFI/centos/grub.cfg

或

grub2-mkconfig -o /boot/efi/EFI/kylin/grub.cfg

或

grub2-mkconfig -o /boot/efi/EFI/uos/grub.cfg

5. 重启服务器

```
reboot
```

6. 执行步骤 1 再次确认

3.1.2. 配置内核参数

配置操作系统内核参数的操作步骤如下：

1. 以 root 用户登录 KINGWOW 服务器；

2. 编辑“/etc/sysctl.conf”文件，在末尾添加以下内容：

```
# for KINGWOW
net.core.somaxconn = 2048
net.core.netdev_max_backlog = 10000
net.core.rmem_default = 16777216
net.core.wmem_default = 16777216
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.ip_local_port_range = 10240 65535
net.ipv4.ip_forward = 0
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
net.ipv4.tcp_syncookies = 0
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 65536 16777216
net.ipv4.tcp_max_syn_backlog = 16384
net.ipv4.tcp_fin_timeout = 15
net.ipv4.tcp_tw_reuse = 1
net.ipv4.tcp_tw_recycle = 1
vm.swappiness = 0
```

3. 执行以下命令，使修改生效；

```
sysctl -p /etc/sysctl.conf
```

3.1.3. 配置时间同步

KINGWOW 数据库系统通常运行于服务器集群之上，对于集群中服务器间的时钟误差有较为严格的要求，可以通过 chrony 时钟同步来保证所有的服务器时间一致。在系统运行期间，如果发生服务器重启的情况，需要关注时钟同步服务是否正常。

使用 NTP 时钟同步需要关闭所有服务器的防火墙和 SELinux(或者开放 NTP 服务相应端口)：

1.关闭防火墙：

service iptables stop

chkconfig iptables off

或

systemctl stop firewalld

systemctl disable firewalld

2. 关闭 SELinux：

执行以下命令检查 SELINUX 的值：

cat /etc/selinux/config |grep "SELINUX="

如果输出结果不是 “SELINUX=disabled” 则执行如下命令修改

```
sed -i s/^SELINUX=.*/SELINUX=disabled/ /etc/selinux/config
setenforce 0
将 SELINUX=enforcing 改为 SELINUX=disabled
```

- **安装 chrony 服务**

```
#安装 chrony
yum install -y chrony
systemctl start chronyd        #启动 chrony 服务
systemctl status chronyd       #查看服务状态
systemctl enable chronyd       #设置开机启动同步时间
```

- **开启 chrony 服务端**

如果没有现成可用的时钟同步服务器，可以在 KINGWOW 数据库集群中任选一台服务器作为时钟同步服务器，在其上开启 chrony 服务，操作步骤如下：

1.以 root 用户登录 NTP Server 服务器；

编辑修改“vi /etc/chrony.conf”文件,主要内容如下 ；

设置 server IP

server {ip} iburst

#根据实际时间计算出服务器增减时间的比率，然后记录到一个文件中，在系统重启后为系统做出最佳时间补偿调整。

driftfile /var/lib/chrony/drift

makestep 1.0 3

将启用一个内核模式，在该模式中，系统时间每 11 分钟会拷贝到实时时钟（RTC）。

rtcsync

指定一台主机、子网，或者网络以允许或拒绝 NTP 连接到时钟服务器的机器

allow all

local stratum 10

指定日志文件的目录。

logdir /var/log/chrony

2. 执行以下命令，重启 chronyd 服务；

systemctl restart chronyd

3. 执行以下命令，设置 NTP 服务开机自动启动；

systemctl enable chronyd

4. 执行以下命令，查看 NTP Server 服务是否启动；

systemctl status chronyd

- **开启 chrony 客户端**

配置文件修改：vim /etc/chrony.conf

修改 server 即可，删掉其他的，添加要同步时间的源服务器 ip，格式如下：

server x.x.x.x iburst

重启 chronyd 服务并设为开机启动，开启网络时间同步功能

systemctl enable chronyd

systemctl restart chronyd

timedatectl set-ntp true

- chrony 常用命令

```
查看时间同步源：chronyc sources -v
立即手工同步：chronyc -a makestep
查看时间同步源状态：chronyc sourcestats -v
设置硬件时间，硬件时间默认为
UTC：timedatectl set-local-rtc 1
#将你的硬件时钟设置为本地时区
设置当前系统为 Asia/Shanghai 上海时区：
timedatectl set-timezone Asia/Shanghai
启用 NTP 时间同步：timedatectl set-ntp yes
校准时间服务器：chronyc tracking
```

最后需要注意的是，配置完/etc/chrony.conf 后，需重启 chrony 服务，否则可能会不生效。

3.1.4. 配置资源限制

Linux 操作系统默认所使用堆栈和文件数的限制值较小，为了正常运行 KINGWOW 数据库服务，需要放大对资源的限制，操作步骤如下：

以 root 用户登录服务器；

配置文件”/etc/security/limits.conf”修改添加如下信息：

```
echo "# for kingwow
```

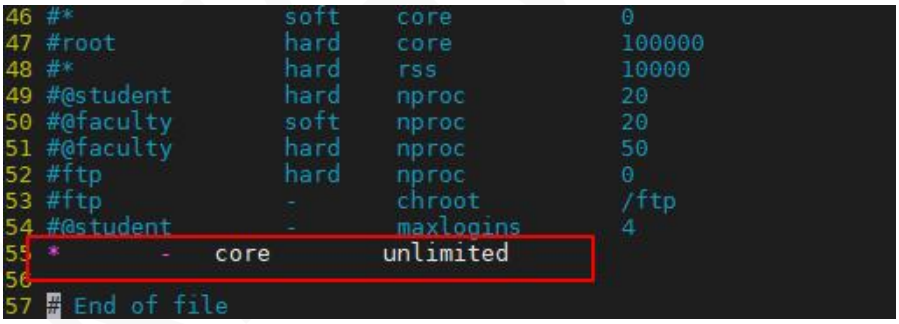
```
admin soft stack 819200

admin hard stack 819200

admin soft nofile 655350

admin hard nofile 655350

admin soft nproc 655360

admin hard nproc 655360

# end

" >>  /etc/security/limits.conf
```

其中 admin 为 KINGWOW 安装用户名

注意：UOS 系统可以通过添加对 core 文件的大小添加无限制来解决可能出现的因 core 文件太大而被截断的问题。被截断后的 core 文件可能无法查看到信息。

操作步骤步骤：

1. 修改/etc/security/limits.conf，在里面设置上 core 文件大小无限制



添加命令

```
echo "

admin - core unlimited

" >>  /etc/security/limits.conf
```

3. 重启机器，重启机器后建议设置下 core 文件生成路径和格式，方便查找。推荐创建单独目录用于存放 core 文件，命令如下：

```
# 创建 core 文件目录 并添加权限
mkdir /kingwow_core
chmod 777 /kingwow_core
# 配置 core 目录
echo "#core_pattern
kernel.core_pattern = /kingwow_core/core-%e-%p-%t
" >>  /etc/sysctl.conf
sysctl -p /etc/sysctl.conf
```

3.2. 创建安装用户

KINGWOW 默认安装用户为 admin,在所有 KINGWOW 服务器中创建安装用户的操作步骤如下：

1. 以 root 用户登录服务器；
2. 执行如下命令，创建安装用户 admin；

   `useradd admin`

3. 执行如下命令，为用户 admin 设置密码；

   `passwd admin`

4. 为用户 admin 赋予 sudo 权限（非必须）；

   直接执行以下命令添加安装用户的 sudo 权限：

   ```
   echo "admin ALL=(ALL) ALL" | sudo tee -a /etc/sudoers
   ```

   或者按以下步骤添加安装用户的 sudo 权限。

   a) 执行以下命令，添加“/etc/sudoers”文件的写权限

      `chmod u+w /etc/sudoers`

   b) 编辑“/etc/sudoers”文件中“root ALL=(ALL) ALL”后添加如下黑体语句：

```
root ALL=(ALL) ALL
admin ALL=(ALL) ALL
```

c) 执行以下命令，删除“/etc/sudoers”文件的写权限；

```
chmod u-w /etc/sudoers
```

3.3. 配置环境变量

为了正常运行 KINGWOW 数据库系统，需要对安装用户 admin 的环境变量进行配置，以 admin 用户登录 KINGWOW 服务器，以数据库程序安装目录为 /home/admin/kingwow 为例操作步骤如下：

1. **命令一键执行：**

```
echo "# for kingwow
export KW_HOME=/home/admin/kingwow
export
PATH=\$KW_HOME/bin:\$KW_HOME/dependlib:\$KW_HOME/tools/bin:\$KW_H
OME/bin:/usr/local/sbin:/usr/sbin:$PATH
export
LD_LIBRARY_PATH=\$KW_HOME/lib:/usr/local/lib64:/usr/:/usr/lib/:/u
sr/local/lib:/lib:\$KW_HOME/dependlib:\$KW_HOME/tools/lib:
ulimit -c unlimited
set -o vi
" >> "/home/admin/.bashrc"
```

手动编辑修改

修改“/home/admin/.bashrc”文件，添加如下语句：

```
export
LD_LIBRARY_PATH=~/kingwow/lib:/usr/local/lib64:/usr/:/usr/lib/:/u
sr/local/lib:/lib
export PATH=~/kingwow/bin:$PATH
```

```
ulimit -c unlimited
```

2. 执行以下命令，让环境配置生效；

```
source ~/.bashrc
```

3.4. 配置服务器间免登录

在所有的 KINGWOW 服务器中选择一台作为本机，配置该服务器到所有服务器的免登录（包括本机到本机）。配置免登录后，该服务器在连接其他服务器时无需输入密码。

3.4.1. 配置过程

假设本机的 IP 地址为“192.168.1.4”，配置免登录的操作步骤如下：

1. 以 admin 用户登录服务器；
2. 执行以下命令，并按“Enter”键，直至生成公钥；

```
ssh-keygen -t rsa -P ''
```

3. 执行以下命令，并根据提示输入登录密码，配置免登录；

```
ssh-copy-id admin@192.168.1.2
ssh-copy-id admin@192.168.1.3
ssh-copy-id admin@192.168.1.4
ssh-copy-id admin@192.168.1.5
```

3.4.2. 验证

配置免登录完成后，在本机中输入“ssh admin@192.168.1.X”；

- 如果无需输入密码，则表示配置免登录成功；
- 如果仍需要输入密码，则请重新配置免登录或关闭 SELINUX 后重试；

4. RPM 包安装

4.1. 复制安装包

将 KINGWOW 数据库的安装文件复制到安装用户 admin 的”~/install”目录下；

4.2. 安装 KINGWOW 数据库

执行以下命令，安装 KINGWOW 数据库：

```
cd ~/install/
sudo rpm -ivh kingwow-5.0.11-20250304.x86_64.rpm
--prefix=/home/admin/kingwow --nodeps
```

注：如果不加 prefix 参数，默认安装在/home/admin/kingwow 目录下；

4.3. 创建数据目录

本样例安装挂载使用了 4 块数据盘到/data/{1..4}，并将目录的所有者修改成 admin:admin

数据目录：~/kingwow/data 最终结构如下

```
data
├── 1 -> /data/1/admindata/dg_data
├── 2 -> /data/2/admindata/dg_data
├── 3 -> /data/3/admindata/dg_data
├── 4 -> /data/4/admindata/dg_data
├── mg
├── mg_commitlog -> /data/1/admindata/commitlog/mg_commitlog
├── tg_commitlog -> /data/1/admindata/commitlog/tg_commitlog
└── tg_data
    ├── raid0
    │   ├── store0 -> /data/1/admindata/tg_data/1
    │   └── store1 -> /data/2/admindata/tg_data/2
    └── raid1
```

```
├── store0 -> /data/3/admindata/tg_data/3
└── store1 -> /data/4/admindata/tg_data/4
```

1. 以 admin 用户登录服务器，并进入安装目录~/kingwow;
2. 执行以下命令，创建和配置所需数据目录:

```
sudo mkdir /data -p
sudo chown admin:admin /data
sh $KW_HOME/scripts/install/RebuildDir.sh -m performence -u 2 -c 4 -d kwtest -s $KW_HOME
```

参数含义:

-m: 创建数据目录的模式：[test|performence] (performance 使用 /data/1..n 作为数据挂载点，test 使用~/data 作为数据挂载点，默认 test)

-u: MG/TG 服务所使用的挂载磁盘数目(默认 4)

-c: KG/KG 服务所使用的挂载磁盘数目(默认 4)

-d: 应用名(默认为 kwtest)

注:脚本随安装文件提供，sh $KW_HOME/scripts/install/RebuildDir.sh -h 查看更多参数选择;

5. 启动与管理系统

在每台服务器上正确安装 Kingwow 软件并且创建数据目录之后，就可以启动数据库服务。

5.1. 数据库服务启动命令

进入 KINGWOW 安装目录：cd $KW_HOME

5.1.1. MG 启动命令

主 MG 启动命令格式:

```
bin/manEng -r 192.168.1.2:2500 -R 192.168.1.2:2500 -i eth0 -U 3 -u '3' -C 0 -K 1 -G 1
```

备 MG 启动命令格式:

```
bin/manEng -r 192.168.1.3:2500 -R 192.168.1.2:2500 -i eth0 -C 0
```

参数含义：

-r：所启动 MG 服务的 IP 地址和端口；

-R：系统主 MG 服务的 IP 地址和端口；

-i：绑定的网卡名；

-C：系统中所属逻辑集群 ID，必须为数字；V5.* 版本后 -C 需要从 0 开始，V3 版本及以前，从 1 开始

-U：指定系统中 MG 服务的数量；

-u：指定系统中每个 paxos 组 TG 服务的数量，每个 group 可以设置不同的值，格式为"x,x,x,x,x,x,x")，中间可以用逗号或者分号隔开

-s：指定备 MG 以 slave_only 模式启动，slave_only 模式的 MG 不会被选为主 MG；

-G：指定 paxos 组个数，必须为数字；

-K：指定 cluster 个数，必须为数字；

5.1.2. TG 启动命令

命令格式：

```
bin/transEng -r 192.168.1.2:2500 -p 2700 -m 2701 -i eth0 -C 0 -g 0
```

参数含义：

-r：系统主 MG 服务的 IP 地址和端口；

-p：所启动 TG 服务的端口号；

-m：所启动 TG 服务的内部端口号；

-i：绑定的网卡名；

-C：系统中所属逻辑集群 ID，必须为数字；V5.* 版本后 -C 需要从 0 开始，V3 版本及以前，从 1 开始

-g：paxos 组 id，必须为数字；与 MG 启动参数-G 参数对应，从 0 开始；

5.1.3. KG 启动命令

命令格式：

```
bin/KwowEng -r 192.168.1.2:2500 -p 2800 -z 2800 -S 2600 -n kwtest -i eth0 -C 0
```

参数含义：

-r：系统主 MG 服务的 IP 地址和端口；

-p：所启动 KG 服务的端口号；

-z：所启动 KG 的数据库服务端口号；

-S：所启动 KG 数据服务的端口号；

-n：指定应用名，应当与建 KG 数据目录时所设定的应用名一致；

-i：绑定的网卡名；

-C：系统中所属逻辑集群 ID，必须为数字；V5.* 版本后-C 需要从 0 开始，V3 版本及以前，从 1 开始

5.2. 单集群启动示例

以一个逻辑集群组成的 KINGWOW 数据库系统为例，集群号为 0，使用三台服务器，在每台服务器上都启动 MG、TG、KG 服务。指定主 MG 为 192.168.1.2:2500。

以 admin 用户登录所有服务器，在安装目录/home/admin/kingwow 下执行启动命令。

Server 1（主 MG 所在服务器）

```
bin/manEng -r 192.168.1.2:2500 -R 192.168.1.2:2500 -i eth0 -U 3 -u '3' -C 0 -K 1 -G 1
```

```
bin/transEng -r 192.168.1.2:2500 -p 2700 -m 2701 -i eth0 -C 0 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 0
```

Server 2

```
bin/manEng -r 192.168.1.3:2500 -R 192.168.1.2:2500 -i eth0 -C 0 -K 1
```

```
bin/transEng -r 192.168.1.2:2500 -p 2700 -m 2701 -i eth0 -C 0 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 0
```

Server 3

```
bin/manEng -r 192.168.1.4:2500 -R 192.168.1.2:2500 -i eth0 -C 0 -K 1
```

```
bin/transEng -r 192.168.1.2:2500 -p 2700 -m 2701 -i eth0 -C 0 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 0
```

如果有其它服务器加入到集群中，可以在这些服务器上启动 KG 服务，命令如下：

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 0
```

注：MG 和 TG 已经达到设定数目；

在所有服务器上执行完毕命令后，确认所有服务进程都已启动。

在任意一台服务的安装目录下执行以下命令确认所有角色均在线，且已选出主 TG:

```
bin/mg_admin -r 192.168.1.2 -p 2500  stat -o all_server
```

确认点，状态正常必须同时满足下述两个条件：

1 输出各类服务的节点和规划启动的一致

2 transEngs 角色有一台的 stat:master，其余为 stat:sync

若 stat 报错-12 或者上述条件未同时满足，请等待数秒后，重新执行 stat 命令。

确认启动正常后，在任意一台服务的安装目录下执行以下命令进行数据库系统的初始化：

```
bin/mg_admin -r 192.168.1.2 -p 2500  -t 600000000 boot_strap
```

5.3. 多集群启动示例

以三个逻辑集群组成的 KINGWOW 数据库系统为例，集群号分别为 1、2、3，可以分别部署在三个机房中。每个逻辑集群使用一台服务器启动 MG、TG、KG 共四个服务，指定主 MG 为 192.168.1.2:2500。

以 admin 用户登录所有服务器，在安装目录/home/admin/kingwow 下执行启动命令。

集群 1：

Server1

```
bin/manEng -r 192.168.1.2:2500 -R 192.168.1.2:2500 -i bond0 -U 3 -u '3' -C 0 -K 3 -G 1
```

```
bin/transEng -r 192.168.1.2: 2500 -p 2700 -m 2701 -i bond0 -C 0 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 0
```

集群 2:

Server2

```
bin/manEng -r 192.168.1.3: 2500 -R 192.168.1.2:2500 -i bond0 -C 1 -K 3
```

```
bin/transEng -r 192.168.1.2: 2500 -p 2700 -m 2701 -i bond0 -C 1 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 1
```

集群 3:

Server3

```
bin/manEng -r 192.168.1.4:2500 -R 192.168.1.2:2500 -i bond0 -C 2 -K 3
```

```
bin/transEng -r 192.168.1.2:2500 -p 2700 -m 2701 -i bond0 -C 2 -g 0
```

```
bin/KwowEng -r 192.168.1.2:2500 -p2800 -z 2880 -S 2600 -n kwtest -i eth0 -C 2
```

在所有服务器上执行完毕命令后，在确认所有服务进程都已启动。

在任意一台服务的安装目录下执行以下命令确认所有角色均在线，且已选出主 TG:

```
bin/mg_admin -r 192.168.1.2 -p 2500  stat -o all_server
```

确认点，状态正常必须同时满足下述两个条件:

1 输出各类服务的节点和规划启动的一致

2 transEngs 角色有一台的 stat:master，其余为 stat:sync

若 stat 报错-12 或者上述条件未同时满足，请等待数秒后，重新执行 stat

命令。

确认启动正常后，在任意一台服务的安装目录下执行以下命令进行数据库系统的初始化：

```
bin/mg_admin -r 192.168.1.2 -p 2500  -t 600000000 boot_strap
```

5.4. 使用 KINGWOW 客户端

在数据库系统初始化完成后，可以执行以下命令连接 KINGWOW 数据库：

```
kingwow -h192.168.1.2 -P2880 -uadmin -padmin
```

参数含义：

-h：KG 所在服务器的 IP 地址；

-P：启动 KG 服务命令-z 参数所指定的数据库服务端口号；

-u：KINGWOW 数据库用户名，admin 为预置超级用户；

-p：KINGWOW 数据库用户的密码，admin 用户的缺省密码为 admin

5.5. 创建图形化客户端需要的视图

执行安装目录下的 SQL 文件，初始化数据库。

```
kingwow -h192.168.1.2 -P2880 -uadmin -padmin -e"source $KW_HOME/scripts/install/view.sql";
kingwow -h192.168.1.2 -P2880 -uadmin -padmin -e"source $KW_HOME/scripts/install/init_parameters_6.X.sql";
kingwow -h192.168.1.2 -P2880 -uadmin -padmin -e"source $KW_HOME/scripts/install/init_privilege.sql";
```

5.6. 配置 crontab 清理数据库垃圾文件及日志文件

使用安装目录下的 scripts 里的相关文件和配置，可以手动修改配置 crontab 定时任务，以下为示例：

设置 crontab 定时任务 `crontab -e`

```
# for kingwow
# 1 点执行定时清理任务
```

```
0 1 * * * source ~/.bashrc && bash $KW_HOME/scripts/file_expire/file_expire_v2.sh >> /kingwow/db/log/file_expire_v2.log 2>&1

#mg,tg,dg,cg 进程自动重启，建议时间间隔为 1 分钟

* * * * * source ~/.bashrc && bash $KW_HOME/scripts/auto_boot/auto_boot.sh >> /kingwow/db/log/auto_boot.log 2>&1

#数据迁移临时清理任务

*/15 * * * * source ~/.bashrc && rm -f $KW_HOME/{1..6}/Recycle/* > /dev/null 2>&1

*/15 * * * * source ~/.bashrc && rm -f $KW_HOME/data/tg_data/raid*/store*/trash/* > /dev/null 2>&1
```

注意：crontab 任务中的配置文件为 $KW_HOME/scripts/auto_boot/auto_boot.config.sh 和 $KW_HOME/scripts/file_expire/file_expire_v2.config.sh,需根据实际情况进行修改。如果提示用户无 crontab 权限,需添加普通用户使用 crontab 的权限,命令如下

```
echo "admin" >> /etc/cron.allow

chmod u+s /usr/bin/crontab
```