TDengine部署

部署TDengine需要干净的基础环境！！！如果之前安装过其他产品，需要重新安装操作系统

服务端目录规划示例：

路径	用途	对应参数
/data/taos/data	数据目录，用于存放数据库数据文件	dataDir
/data/taos/log	日志目录，用于存放数据库日志	logDir
/data/taos/core	用于存放数据库异常时产生的core文件	kernel.core_pattern
/data/taos/tmp	用于存放数据库运行时产生的临时文件	tempDir
/data/taos/soft	用于存放安装包	
/data/taos/dump	备份存放目录	

注意：生产环境不建议将数据目录与日志目录放在同一个磁盘上，以避免IO竞争。

客户端目录规划示例：

路径	用途	对应参数
/data/taos/log	日志目录，用于存放客户端日志	logDir
/data/taos/core	用于存放客户端异常时产生的core文件	kernel.core_pattern
/data/taos/tmp	用于存放数据库运行时产生的临时文件	tempDir
/data/taos/soft	用于存放安装包	

1.环境准备

操作系统安装建议：

1.采用最小化安装

2.安装时设置 SWAP、时区、主机名

SWAP设置可参考 What is the recommended swap size for Red Hat platforms?

3.文件系统建议选用ext4，格式化是建议配置【lazy_itable_init=0,lazy_journal_init=0】，挂载时加上【data=ordered】参数。

4.如果数据量大，磁盘空间大，建议选择 xfs。

> 实验测试结果：
>
> 在写入方面xfs优于ext4[未优化]，速度提升约25%；关闭barrier后，ext4写入速度较xfs提升近50%。

> 在查询方面，数据离散度越大，ext4表现越好，速度提升约25%；而对于较为连续的查询，ext4速度反而不如xfs。

1.1.准备安装目录

按照前期规划，创建相关目录。生产环境中数据目录和日志目录建议部署在不同磁盘上。

```
mkdir -p /data/taos/{data,log,core,soft,tmp,dump}
mkdir -p /data/taos/{log,core,soft,tmp}
```

1.2.安装软件包

为方便 TDengine 运维及 debug，建议安装以下辅助工具包。

```
#CentOS
yum install -y screen
yum install -y tmux
yum install -y gdb
yum install -y fio
yum install -y iperf3
yum install -y sysstat
yum install -y net-tools
yum install -y ntp
yum install -y tree
yum install -y wget
wget https://repos.baslab.org/rhel/7/bpftools/bpftools.repo -O
/etc/yum.repos.d/bpftools.repo --no-check-certificate
yum install -y bpftrace

#Ubuntu
apt install -y screen
apt install -y tmux
apt install -y gdb
apt install -y fio
apt install -y iperf3
apt install -y sysstat
apt install -y net-tools
apt install -y ntp
apt install -y tree
apt install -y wget
```

工具包说明：

screen | tmux：终端管理软件，可以进行分屏，有效防止远程操作意外中断情况。

gdb：软件调试工具。

fio：磁盘性能测试工具。

iper3：网络性能测试工具。

sysstat：常用系统资源查看工具包。

net-tools：常用网络工具包。

ntp：时间同步工具。

tree：目录结果可视化工具。

bpftrace：程序性能诊断工具。

1.3.配置域名解析

TDengine 节点间通信，依赖于域名解析，需保证各节点间能够解析所有节点地址。在所有节点配置如下：

```
192.168.0.231 td1.taosdata.com
192.168.0.232 td2.taosdata.com
192.168.0.233 td3.taosdata.com
192.168.0.234 cl1.taosdata.com
```

1.4.关闭 SELinux

SELinux 是 RedHat 开发的一套安全增强工具，建议关闭以防出现未知问题。

```
sed -i 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/selinux/config
```

1.5.关闭防火墙

TDengine 节点间通信采用 TCP 和 UDP 协议，并使用 6030~6040 端口，同时使用 6041 和 6060 对外提供 RESTful 和 Web 管理服务。

建议关闭防火墙，如需开启防火墙请开放相关端口。关闭防火墙命令如下：

```
#CentOS
systemctl stop firewalld
systemctl disable firewalld

#Ubuntu
ufw stop
ufw disable
```

1.6.配置资源限制

为保证 TDengine 运行时获取到足够的系统资源，需要对配置相应的资源限制。

```
echo "fs.nr_open = 1048576" >>/etc/sysctl.conf
sysctl -p

echo "* soft nproc  65536" >>/etc/security/limits.conf
echo "* soft nofile 65536" >>/etc/security/limits.conf
echo "* soft stack  65536" >>/etc/security/limits.conf
echo "* hard nproc  65536" >>/etc/security/limits.conf
echo "* hard nofile 65536" >>/etc/security/limits.conf
echo "* hard stack  65536" >>/etc/security/limits.conf

echo "root soft nproc  65536" >>/etc/security/limits.conf
echo "root soft nofile 65536" >>/etc/security/limits.conf
echo "root soft stack  65536" >>/etc/security/limits.conf
echo "root hard nproc  65536" >>/etc/security/limits.conf
echo "root hard nofile 65536" >>/etc/security/limits.conf
echo "root hard stack  65536" >>/etc/security/limits.conf
```

1.7.配置时区

如果在配置文件 taos.cfg 中没有配置时区，TDengine 默认采用操作系统设置，建议操作系统和 TDengine 时区保持一致。

操作系统时区设置：以东八区为例

```
cp -f /usr/share/zoneinfo/Asia/Shanghai /etc/localtime
```

1.8.配置 NTP

TDengine 节点间时间必须保证同步，否则会造成节点间状态异常，建议配置 NTP 服务。

如无法连接互联网，建议在局域网内部部署 NTP 服务器。

```
#CentOS 6/7
systemctl start ntpd
systemctl enable ntpd

ntpq -p

#Ubuntu/CentOS 8+
systemctl start  chronyd
systemctl enable chronyd

chrony sourcestats -v
```

1.9.设置 SWAP

设置 SWAP 可以有效降低程序被 OOM 的概率，在操作系统安装时可设置SWAP分区，如果当时没有设置，或设置的过小，可以通过以下方法手动设置：

```
dd if=/dev/zero of=/data/swapfile bs=1M count=4096
mkswap /data/swapfile
swapon /data/swapfile

vi /etc/fstab
/data/swapfile swap                    swap    defaults        0 0
```

1.10.设置 coredump 目录

TDengine 运行异常时，会生成 coredump 文件，该文件可以帮助快速定位问题。但通常 coredump 文件体积较大，建议放置在单独目录下。

```
echo "ulimit -c unlimited" >>/etc/profile
echo "kernel.core_pattern=/data/taos/core/core-%e-%p" >>/etc/sysctl.conf
sysctl -p
```

```
set_core /data/taos/core
```

1.11.关闭 NUMA

NUMA 是一种新型的 CPU 使用内存的架构模型，不适用于 TDengine 应用场景，建议关闭。

在启动配置文件添加 numa=off，以关闭 NUMA。

```
vi /etc/default/grub
GRUB_CMDLINE_LINUX="... numa=off"

grub2-mkconfig -o /etc/grub2.cfg
```

1.12.重启服务器

以上多个配置项的修改需要重启服务器才能生效。

```
reboot
```

1.13.环境验证

服务器重启完成后，对如上修改进行验证。验证命令如下：

```
date -R
tree /data
getenforce
systemctl status firewalld
sysctl -a | grep -E "nr_open|core_pattern"
ulimit -a | grpe core
rpm -qa| grep -E -w "tmux|gdb|fio|iper3|sysstat|net-tools|ntp|tree|bpftrace"
free -m
lscpu| grep 'NUMA node(s):'
```

建议执行环境检查脚本 preCheck.sh

1.14.基础性能验证

网络健康状态检查，iperf 工具可以检查带宽和丢包率，需要在网络两端分别启动服务端和客户端。

```
iperf3 -s  ###启动iperf服务端
iperf3 -c  ###启动iperf客户端
```

磁盘读写性能检查

```
fio -ioengine=libaio -direct=1 --iodepth=32 -thread -rw=randwrite -
filename=/data/test -runtime=60 -numjobs=4 -filesize=20g -bsrange=480k-500k -loop
1000 -name="Test"
```

```

```

2.部署 TDengine 集群

环境配置完成后，即可进行 TDengine 安装，将 TDengine 安装包放置在各服务器 /data/taos/soft/ 目录下。

2.1.安装 TDengine 服务端

TDengine 服务端需要安装两个安装包，分别为：

安装包	说明
TDengine-enterprise-server-2.4.0.7-Linux-x64.tar.gz	TDengine 主程序包，包含数据库主程序 taosd 和 RESTful 接口程序 taosAdapter。
taosTools-1.3.0-Linux-x64.tar.gz	TDengine 工具包，包括 Benchmark 工具和备份工具。

2.1.1.安装 TDengine 服务端

```

```

```
cd /data/taos/soft/
tar xvzf TDengine-enterprise-server-2.4.0.7-Linux-x64.tar.gz
cd TDengine-enterprise-server-2.4.0.7
./install.sh -e no
```

TDengine 服务端安装完成后会自动创建名为 taosd 和 taosAdapter 的服务，并将 taosd 服务设置为开机自启动。

2.1.2.安装 TDengine 工具包

```
cd /data/taos/soft/
tar xvzf taosTools-1.3.0-Linux-x64.tar.gz
cd taosTools-1.3.0
./install-taostools.sh
```

2.1.3.修改配置文件 taos.cfg

默认的配置文件中，所有参数都是注释掉的，而且参数非常多，为方便后期维护，建议备份默认配置文件，重新创建一份新的配置文件。如 taosd 进程使用非 root 用户启动，需要保证 taosd 进程拥有读取 taos.cfg 文件权限。

备份默认配置文件

```
cd /etc/taos
mv taos.cfg taos.cfg.bak
```

创建新配置文件 taos.cfg，内容如下：

2.6

```
#activeCode
firstEp                    node1:6030
secondEp                   node2:6030
fqdn                       node1
arbitrator                 node2:6042
logDir                     /data/taos/log
dataDir                    /data/taos/data
tempDir                    /data/taos/tmp
numOfThreadsPerCore        2.0
ratioOfQueryCores          2.0
numOfCommitThreads         8.0
minTablesPerVnode          1000
tableIncStepPerVnode       1000
maxVgroupsPerDb            8
keepColumnName             1
```

```
balance                       0
blocks                        6
maxSQLLength                  1048576
maxNumOfOrderedRes            100000
maxNumOfDistinctRes           10000000
maxWildCardsLength            100
update                        2
cachelast                     1
timezone                      UTC-8
locale                        en_US.UTF-8
charset                       UTF-8
maxShellConns                 100000
maxConnections                100000
monitor                       1
logKeepDays                   -1
debugflag                     131
rpcForceTcp                   1
slaveQuery                    0
numOfMnodes                   3
offlineInterval               15
tcpConnTimeout                100
telemetryReporting            0
#shellActivityTimer           120
#compressMsgSize              -1
#compressColData              -1
#keepTimeOffset               0
#queryRssThreshold            4096
```

```
shellActivityTimer        120
firstEp                   node1:6030
secondEp                  node2:6030
fqdn                      node
maxShellConns             100000
maxConnections            100000
maxNumOfDistinctRes         10000000
logDir                    /var/log/taos
dataDir                   /var/lib/taos
tempDir                   /tmp/
supportVnodes             0
minSlidingTime            10
minIntervalTime           10
queryBufferSize           -1
compressMsgSize           -1
compressColData           -1
timezone                  UTC-8
locale                    en_US.UTF-8
charset                   UTF-8
monitor                   1
logKeepDays               -1
countAlwaysReturnValue    1
```

```
numOfCommitThreads        8
```

3.0

```
#activeCode
firstEp                   node1:6030
secondEp                  node2:6030
fqdn                      node1
logDir                    /data/taos/log
dataDir                   /data/taos/data
tempDir                   /data/taos/tmp
numOfCommitThreads        8.0
numOfVnodeQueryThreads    8
timezone                  UTC-8
locale                    en_US.UTF-8
charset                   UTF-8
monitor                   1
monitorFqdn               localhost
logKeepDays               -1
debugflag                 131
keepColumnName            1
countAlwaysReturnValue    1
supportVnodes             0
slowLogThreshold          10
```

参数解读：

见官方文档

> 参数设置注意事项：
>
> offlineInterval > tcpConnTimeout * numOfCommitThreads
>
> numOfCommitThreads = 当前节点vnode数量
>
> maxVgroupsPerDb = min{64/dnods_num, cpu_num * 0.6}
>
> compressMsgSize，compressColData 仅适用于网络带宽不足的环境
>
> keepTimeOffset = 期望迁移时间-8
>
> queryRssThreshold 限制taosd 内存使用，单位MB，如果超过阈值，无法建立新连接。 需要和 MALLOC_CONF=background_thread:true 配合使用
>
> numOfVnodeQueryThreads = vCPU - numOfCommitThreads

如果使用taosAdapter模块，需要指定其目录位置，并根据实际情况修改连接池数量（CPU*2）。

/etc/taos/taosadapter.toml

```
debug = false

[pool]
maxConnect = 32
maxIdle = 32

[log]
path = "/data/taos/log"
```

```
## 2.6
[monitor]
writeToTD = true
user = "root"
password = "New_Password"

## 3.0
[monitor]
writeToTD = false
```

2.1.4.启动数据库

```
systemctl start taosd
###检查服务状态
systemctl status taosd
```

如果使用taosAdapter模块，启动命令如下：

```
systemctl start taosadapter
###检查服务状态
systemctl status taosadapter
###设置开机自启动
systemctl enable taosadapter
```

2.2.创建 TDengine 集群

TDengine支持水平动态扩展，单个 TDengine 节点即可视为 1 个节点的 TDengine 集群。

以下操作会将 td2 和 td3 加入原 td1 集群，构成一套 3 节点的 TDengine 集群。

2.2.1.安装 TDengine 服务端

根据2.1 安装方法，在 td2 和 td3 安装 TDengine 服务端和 TDengine 工具包。

2.2.2.修改配置文件 taos.cfg

在 td2 和 td3 节点，分配修改配置文件如下：

```
如上：
```

配置完成后，分别在 td2 和 td3 节点启动 TDengine 服务。

```
systemctl start taosd
###检查服务状态
systemctl status taosd
```

2.2.3.将新节点加入集群

在 td1 上执行 taos 命令，进入TDengine 客户端，执行添加节点命令如下：

```
CREATE DNODE  "td2.taosdata.com:6030";
CREATE DNODE  "td3.taosdata.com:6030";
```

查看数据节点和管理节点状态，如果状态非offline，则表明添加成功。

```
SHOW DNODES;
SHOW MNODES;
```

3.0 创建mnode

```
CREATE MNODES ON DNODE 2;
CREATE MNODES ON DNODE 3;
```

3.创建数据库

执行 taos 命令，进入TDengine 客户端，执行建库命令：

2.x

```
CREATE DATABASE db12
DAYS 1
BLOCKS 12
REPLICA 3
KEEP 7,30,3650
MINROWS 100
```

```
MAXROWS 4096
UPDATE 2;
```

3.0

```
CREATE DATABASE db12
CACHESIZE   1024  --MB/per Vnode
CACHEMODEL "both"
DURATTION 10  --day
BUFFER 128  --MB/per Vnode
VGROUPS 12
REPLICA 3
KEEP 7,30,3650
MINROWS 100
MAXROWS 4096
STT_TRIGGER 4
PAGES 512
PAGESIZE 8;  --KB
```

3.1.修改root密码

```
alter user root pass 'Tbase2024_0707###';
```

3.2.创建用户

创建监控用户

```
create user monitor pass 'Tbase_Mon_2024###'
```

创建只读用户

```
create user test pass 'test' sysinfo 0;
//2.6
alter user test privilege read;
//3.0
grant read on db.* to test;
```

3.3.修改taosKeeper配置

/etc/taos/taoskeeper.toml

```

```

```
username = "monitor"
password = "Tbase_Mon_2024###"
```

```
systemctl start  taoskeeper
systemctl enable taoskeeper
```

4.生产环境最佳实践(仅适用于2.0)

视具体业务场景而定。

4.1.内存优化

优化内存回收速度。 修改 taosd 启动文件 /etc/systemd/system/taosd.service 在 [Service] 中添加

```
Environment="MALLOC_CONF=background_thread:true"
```

4.2.关闭速度优化

对于大数据量高并发场景，由于大量 WAL 数据需要落盘，会造成 taosd 关闭时耗时较长。 如需要加快关闭服务速度，进行以下操作：

```
sync && sync && systemctl stop taosd & kill -9 `pidof taosd`
```

4.3.单节点重启步骤

如果要重启集群中的某个节点，或某个节点意外终止需要重新启动，需遵循以下步骤：

a.使用 kill -9 `pidof taosd` 停止taosd服务（前提服务配置文件已优化）。

b.清理所有vnode的wal文件。

c.如果节点停机时间长，数据差异大，可先手动同步vnode下文件。

d.使用 systemctl start taosd 启动服务。

4.4.操作系统优化

/etc/sysctl.conf

```
net.core.somaxconn=10240
net.core.netdev_max_backlog=20480
net.ipv4.tcp_max_syn_backlog=10240
net.ipv4.tcp_retries2=5
net.ipv4.tcp_syn_retries=2
```

```
net.ipv4.tcp_synack_retries=2
net.ipv4.tcp_tw_reuse=1
net.ipv4.tcp_tw_recycle=1
net.ipv4.tcp_keepalive_time=600
net.ipv4.tcp_abort_on_overflow=1
net.ipv4.tcp_max_tw_buckets=5000

net.core.wmem_default=212992
net.core.wmem_max=212992
net.core.rmem_default=212992
net.core.rmem_max=212992
```