

银河麒麟服务器操作系统 V4

HBase 软件适配手册



KYLIN
银河麒麟

天津麒麟信息技术有限公司

2019年5月

目 录

目 录	I
1 概述	2
1.1 系统概述	2
1.2 环境概述	2
1.3 HBASE 软件简介	2
1.4 HBASE 体系架构	2
1) CLIENT	3
2) ZOOKEEPER	3
3) MASTER	3
4) REGIONSERVER	3
5) HLOG(WAL LOG)	3
6) REGION	4
7) MEMSTORE 与 STOREFILE	4
2 HBASE 软件适配	4
2.1 下载 HBASE	4
2.2 配置文件修改	4
2.2.1 配置环境变量	4
2.2.2 修改 HBASE-ENV.SH	5
2.2.3 配置 HBASE-SITE.XML	5
2.2.4 启动 HBASE	5
3 HBASE 常用操作	5
1) 进入 HBASE SHELL	5
2) 创建表	5
3) 查看表详情	5
4) 显示所有的表	6
5) 插入数据	6
6) 查询数据	6
7) 删除表	6
8) 退出 HBASE	6

1 概述

1.1 系统概述

银河麒麟服务器操作系统主要面向军队综合电子信息系统、金融系统以及电力系统等国家关键行业的服务器应用领域，突出高安全性、高可用性、高效数据处理、虚拟化等关键技术优势，针对关键业务构建的丰富高效、安全可靠的功能特性，兼容适配长城、联想、浪潮、华为、曙光等国内主流厂商的服务器整机产品，以及达梦、金仓、神通、南大通用等主要国产数据库和中创、金蝶、东方通等国产中间件，满足虚拟化、云计算和大数据时代，服务器业务对操作系统在性能、安全性及可扩展性等方面的需求，是一款具有高安全、高可用、高可靠、高性能的自主可控服务器操作系统。

1.2 环境概述

服务器型号	长城信安擎天 DF720 服务器
CPU 类型	飞腾 2000+处理器
操作系统版本	Kylin-4.0.2-server-sp2-2000-19050910.Z1
内核版本	4.4.131
HBase 版本	1.4.9

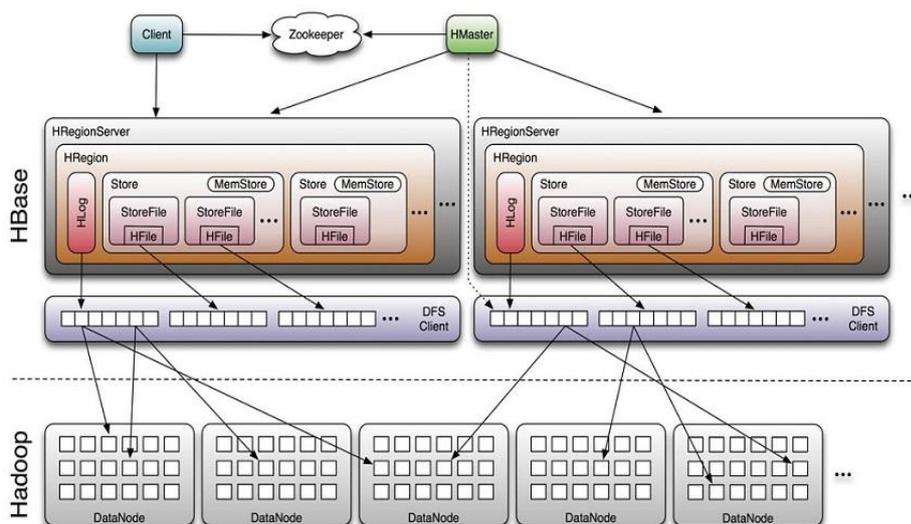
1.3 HBase 软件简介

HBase 是一个分布式的、面向列的开源数据库，该技术来源于 Fay Chang 所撰写的 Google 论文“Bigtable: 一个结构化数据的分布式存储系统”。就像 Bigtable 利用了 Google 文件系统（File System）所提供的分布式数据存储一样，HBase 在 Hadoop 之上提供了类似于 Bigtable 的能力。HBase 是 Apache 的 Hadoop 项目的子项目。HBase 不同于一般的关系数据库，它是一个适合于非结构化数据存储的数据库。另一个不同的是 HBase 基于列的而不是基于行的模式。HBase 具有如下特别：

- 高可靠性、高性能、面向列、可伸缩、实时读写的分布式数据库
- 利用 Hadoop HDFS 作为其文件存储系统，利用 Hadoop MapReduce 来处理 HBase 中的海量数据，利用 Zookeeper 作为其分布式协同服务
- 主要用来存储非结构化和半结构化的松散数据（列存 NoSQL 数据库）

1.4 HBase 体系架构

HBase 体系架构如下图所示：



1) Client

- 包含访问 HBase 的接口并维护 cache 来加快对 HBase 的访问

2) Zookeeper

- 保证任何时候，集群中只有一个 master
- 存贮所有 Region 的寻址入口。
- 实时监控 Region server 的上线和下线信息。并实时通知 Master
- 存储 HBase 的 schema 和 table 元数据

3) Master

- 为 Region server 分配 region
- 负责 Region server 的负载均衡
- 发现失效的 Region server 并重新分配其上的 region
- 管理用户对 table 的增删改操作

4) RegionServer

- Region server 维护 region，处理对这些 region 的 IO 请求
- Region server 负责切分在运行过程中变得过大的 region

5) HLog(WAL log)

- HLog 文件就是一个普通的 Hadoop Sequence File，Sequence File 的 Key 是 HLogKey 对象，HLogKey 中记录了写入数据的归属信息，除了 table 和 region 名字外，同时还包括 sequence number 和 timestamp，timestamp 是”写入时间”，sequence number 的起始值为 0，或者是最近一次存入文

件系 统中 sequence number。

- HLog SequeceFile 的 Value 是 HBase 的 KeyValue 对象，即对应 HFile 中的 KeyValue。

6) Region

- HBase 自动把表水平划分成多个区域(region)，每个 region 会保存一个表里面某段连续的数据；每个表一开始只有一个 region，随着数据不断插入表，region 不断增大，当增大到一个阈值的时候，region 就会等分会 两个新的 region（裂变）；
- 当 table 中的行不断增多，就会有越来越多的 region。这样一张完整的表 被保存在多个 Regionserver 上。

7) Memstore 与 storefile

- 一个 region 由多个 store 组成，一个 store 对应一个 CF（列族）。
- store 包括位于内存中的 memstore 和位于磁盘的 storefile 写操作先写入 memstore，当 memstore 中的数据达到某个阈值，hregionserver 会启动 flashcache 进程写入 storefile，每次写入形成单独的一个 storefile。
- 当 storefile 文件的数量增长到一定阈值后，系统会进行合并(minor、major compaction)，在合并过程中会进行版本合并和删除工作（major），形成更大的 storefile。
- 当一个 region 所有 storefile 的大小和超过一定阈值后，会把当前的 region 分割为两个，并由 hmaster 分配到相应的 regionserver 服务器，实现负载均衡。
- 客户端检索数据，先在 memstore 找，找不到再找 storefile。
- HRegion 是 HBase 中分布式存储和负载均衡的最小单元。最小单元就表示不同的 HRegion 可以分布在不同的 HRegion server 上。
- HRegion 由一个或者多个 Store 组成，每个 store 保存一个 columns family。
- 每个 Store 又由一个 memStore 和 0 至多个 StoreFile 组成。

2 HBase 软件适配

2.1 下载 HBase

```
$ wget http://mirror.bit.edu.cn/apache/hbase/1.4.9/hbase-1.4.9-bin.tar.gz
$ tar xvf hbase-1.4.9-bin.tar.gz -C /usr/local/
```

2.2 配置文件修改

2.2.1 配置环境变量

```
$ vim /etc/profile
```

添加：

```
export HBASE_HOME=/usr/local/hbase-1.4.9
export PATH=$PATH:$HBASE_HOME/bin
```

加载变量：

```
$ source /etc/profile
```

2.2.2 修改 hbase-env.sh

```
$ cd /usr/local/hbase-1.4.9/
$ vim conf/hbase-env.sh
```

添加：

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-arm64
export HBASE_MANAGES_ZK=true
```

2.2.3 配置 hbase-site.xml

```
vim conf/hbase-site.xml
```

内容如下：

```
<configuration>
  <property>
    <name>hbase.rootdir</name>
    <value>file:///data/hbase</value>
  </property>
</configuration>
```

2.2.4 启动 HBase

```
$ ./bin/start-hbase.sh
```

3 HBase 常用操作

1) 进入 hbase shell

```
$ hbase shell
```

2) 创建表

```
hbase(main):001:0> create 'student','Sname','Ssex','Sage','Sdept','course'
hbase(main):002:0> create 'teacher',{NAME=>'username',VERSIONS=>5}
```

3) 查看表详情

```
$ hbase(main):003:0> describe 'student'
```

4) 显示所有的表

```
hbase(main):004:0> list
```

5) 插入数据

```
hbase(main):005:0> put 'student','95001','Sname','LiYing'  
hbase(main):006:0> put 'student','95001','Ssex','Male'  
hbase(main):007:0> put 'student','95001','course:math','80'
```

6) 查询数据

```
hbase(main):008:0> get 'student','95001'  
hbase(main):009:0> get 'student','95001','course'  
hbase(main):010:0> get 'student','95001','course:math'  
hbase(main):011:0> scan 'student'
```

7) 删除表

```
hbase(main):014:0> disable 'student'  
hbase(main):015:0> drop 'student'
```

8) 退出 hbase

```
hbase(main):016:0> exit
```