Linux 集群实时监控系统的一种实现方法①

熊 齐, 唐佳明

(湖南文理学院 国家级 Linux 技术培训与推广中心, 常德 41500)

摘 要: 随着云计算技术的日益发展, Linux 集群以造价低廉、易于扩充等优势得到了愈来愈广泛的应用. 为了更好地发挥集群性能, 充分利用集群节点的资源, 对集群性能进行实时监控是很有必要的. 提出了一种 Linux 集群监控器设计与实现方法. 该方法通过每隔一段时间采集节点机/proc 虚拟文件系统中的信息, 如 CPU 和内存使用情况等. 经过过滤后, 通过 socket 传输给监控服务器. 论文首先给出了监控器的总体设计方案, 整个监控系统由守护在管理节点上的信息管理服务器进程和运行在各个计算节点上的采集器进程组成. 然后分采集器和信息管理器两大部分, 分别介绍了其具体的设计框架和其采用的关键技术. 采集器分主要由信息采集、信息处理和信息传送 3 个模块组成, 分别采用 3 个线程来完成. 信息管理器采用了线程池技术, 用以接受采集器发送过来的传输请求. 实践证明, 该系统可以很好地满足实时监控 Linux 集群性能的需要.

关键词: Linux 集群; 实时; 监控系统; 线程池

An Implementation of Linux Cluster Real-Time Monitoring System

XIONG Qi, TANG Jia-Ming

(National Linux Technology Training & Development Center, Hunan University of Arts and Science, Changde 415000, China)

Abstract: With the rapid development of cloud computing, Linux cluster is used widely with the advantages of low cost and good scalability. This paper presents a method of design and implementation of Linux cluster monitor. Some information, such as the CPU and memory usage, are collected from /proc virtual file system on every node at regular intervals. After being filtered, these information are sent to monitor server via socket. The overall design scheme of monitor is first described in the paper. It is composed by information management server process, which is run on the management node and collecting processes, which are run on every node. Then their design framework and key techniques are introduced.The collector is composed by three modules that is information collection, information processing and information transmission. These modules are realized by three threads. The thread pool technology is used by information management server, which receives the transfer request sent by collector. Proved by practice, this system can well satisfy the real-time monitoring of Linux cluster performance.

Key words: Linux cluster; real-time; monitoring system; thread pool

1 引言

云计算是近年来一个炙手可热的技术名词, 很多专家都认为, 云计算会改变互联网的技术基础, 从而将影响整个产业的格局. 云计算的概念的目前有很多种解释, 一种比较常用的一种解释是, 它是在集群的基础上, 利用虚拟化技术, 使用户可以弹性化地使用集群资源[1]. 当用户使用的集群节点负载过重时, 可以随时申请增加资源. 其关键技术, 就是集群和虚拟化技术. 随着云计算的兴起, 集群系统发展到今天, 节点机的数量已经十分巨大, 几百台节点构成的集群也是很常见的. 不管云计算中的集群提供什么样的服务, 对集群节点各个性能指标的监控必不可少,这样才

① 基金项目:湖南省自然科学基金(12JJ9022);湖南省科技计划(2011GK3185)
收稿时间:2013-02-06;收到修改稿时间:2013-03-18

能充分发挥云计算集群服务器性能.

2 监控系统总体设计

论文设计的集群监控系统由守护在管理节点上的信息管理服务器进程和运行在各个计算节点上的采集器进程组成. 系统监控的流程是:每个计算节点或其上部署的虚拟机的信息由运行在本地的采集器采集后发送给管理节点的信息管理服务器进程,并由其中的信息收集模块收集,最终通过整理后显示出来. 总体整体框架如图 1 所示[2]. 下面介绍系统的主要组成部分以及其实现方法.

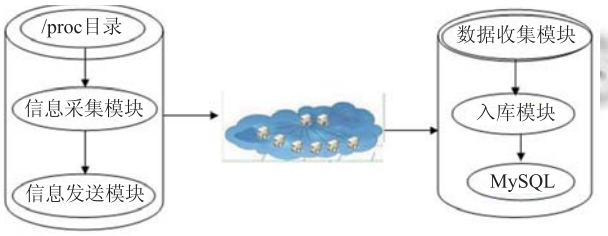

图 1 系统整体框架图

2.1 采集器

采集器是一个部署在集群各计算节点上的信息采集程序,分为信息采集、信息处理、信息发送三个模块. 信息采集主要是抽取 Linux 下/proc 文件系统的部分信息. 与其它常见的文件系统不同的是, /proc 是一种伪文件系统(也即虚拟文件系统), 存储的是当前内核运行状态的一系列特殊文件, 用户可以通过这些文件查看有关系统硬件及当前正在运行进程的信息.

信息处理模块都采集到的信息按照一定的算法进行处理, 从而获得获取系统所需要的监控信息, 比如, 比如对/proc/stat 中获得的信息进行处理可以得出 CPU 占用率, 内存使用量, 内存利用率等信息.

数据发送模块负责把处理好的节点状态信息通过集群间的高速以太网发送给信息管理器.

2.2 信息管理器

信息管理器是整个监控系统的核心,部署在唯一标识的管理节点上,接收来自每个计算节点的采集器发送给它的信息,并将信息整理后存储, 以供信息的发布. 信息管理器主要包括 3 个部分:

● 信息收集:负责接收来自采集器发送至管理节点的信息.

● 信息存储:把收集到的节点信息储存到数据库中.

● 信息发布:以终端形式发布信息,并提供简易的命令方便用户查看和记录信息.

2.3 信息传输方式

监控系统采用“推”的方式来传输信息:采集器只要处于运行状态中就会每隔一段时间采集/proc 文件系统相关信息, 并对这些信息进行处理, 将其不经过任何的压缩“推”送给信息管理服务器. “推”的方式可以保证信息管理服务器收集到实时信息[3].

3 采集器的具体设计

3.1 设计框架

节点机上的采集器主要由信息采集、信息处理和信息传送 3 个模块组成, 分别由 3 个线程来处理完成. 信息采集线程从/proc 文件系统中读取 CPU、内存等系统信息数据放入缓冲区 1 中, 经过信息处理线程分析处理后, 存入缓冲区 2 中, 再由传输线程发送到服务器中. 其框架如图 2 所示.

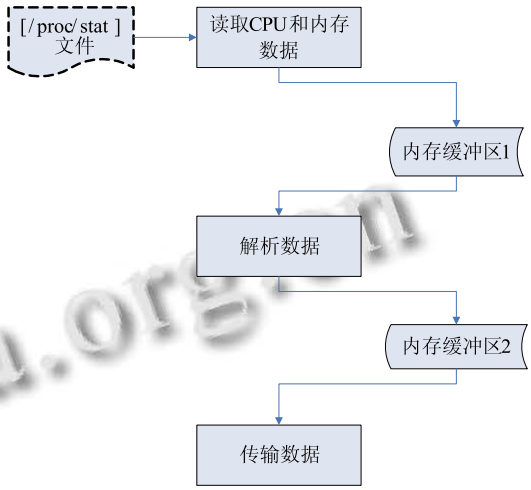

图 2 监控系统客户端的框架图

3.2 关键技术

这里以 CPU 利用率为例, 给出其算法如下:

输入: 两次 CPU 使用情况的数据

输出: CPU 的使用率 usage

数据结构:

采样 CPU 快照, 其结构体为:

```
struct    CPU
{
        user;              //用户
```

```
        nice;          //低优先级用户
        system;        //系统
        idle;          //空闲
        iowait;        //等待输入/输出
        irq;           //中断
        softirq;       //软中断
        stealstolen;
        guest;
}
```

伪代码如下:

```
void  CPUParse (Cpu_t cpuBuf[], float &usage)
{
```

① 采样两个足够短的时间间隔的 CPU 快照，分别记作 t1、t2;

② 计算总的 CPU 时间片 TotalCpuTime

a. 把第一次的所有 CPU 使用情况求和，得到 sum[0].

b. 把第二次的所有 CPU 使用情况求和，得到 sum[1].

c. sum[1] - sum[0]得到这个时间间隔内的所有时间, TotalCpuTime = sum[1] - sum[0].

③ 计算空闲时间 IdleTime

Idle 对应第四列的数据，IdleTime = cpuBuf[1].idle - cpuBuf[0].idle.

④ 计算 CPU 使用率 usage

usage = 100 * (TotalCpuTime - IdleTime) / (float) TotalCpuTime

4 信息管理器的具体设计

4.1 设计框架

管理节点上的信息管理器，利用一个监听线程，接受客户端的连接请求后放入连接缓冲区，再由多个处理线程处理客户端发来的传输请求，并将信息组装成 SQL 语句放入 SQL 缓冲区中，最后由入库线程把信息存入 Mysql 数据库中. 其设计框架如图 3 所示.

4.2 关键技术

信息管理器如果采用传统的 Linux 多线程服务器模型，一旦接受到请求之后，即创建一个新的线程，由该线程接收信息并插入到 SQL 缓存区中. 任务执行完毕后，线程便退出，这就是“即时创建，即时销毁”的方法. 尽管与进程创建相比，线程创建的时间已经大为的缩短，但是如果需要频繁的创建线程，并且每个线程所占用的处理时间又非常简短，则线程创建和销毁将会给处理器带来很多额外的负担[4,5]. 所以我们在这里采取了线程池技术. 在信息管理器采用启动之后，先预创建一定数量的线程，放入空闲队列中，并使其处于阻塞(Suspended)状态，不消耗 CPU 资源，只占用较小的内存空间. 当接到采集器的发送请求后，就从空闲队列中选择一个空闲线程，由该线程负责处理接收任务，同时将其移入忙碌队列. 任务执行完毕后线程并不销毁出，而是被移入空闲队列中等待下一次的任务. 线程池容纳的最大线程数以及最小线程数，可以在配置文件中指定.

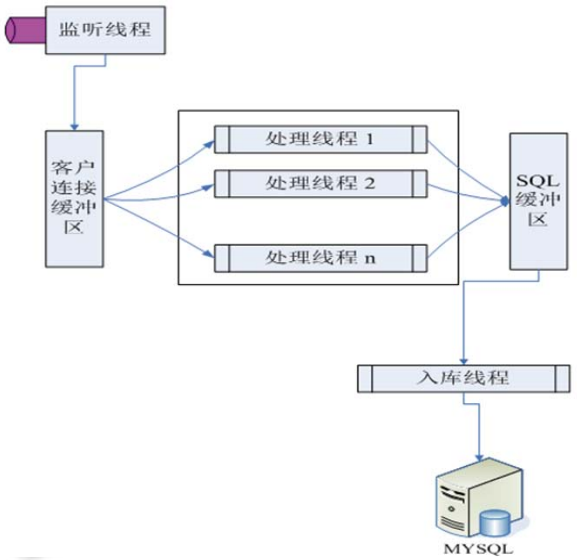

图 3　信息管理器设计框架图

5 运行与测试

5.1 实验环境

实验在湖南文理学院国家 Linux 技术培训与推广中心的环境下，选取了五台 PC 机作为集群环境，其中一台作为管理节点，其余四台作为计算节点. 机器配置:处理器为 Pentium 双核 CPU 3.20 GHz ,内存为 2GB ,硬盘为 150 GB ,网络为 100 Mb/ s 以太网,节点操作系统为 Asianux Server 4 ,数据库为 MySQL5.5.21.编译环境 g++.

5.2 运行结果

管理节点上的信息管理器运行后，便其处于监听状态，当节点机运行后，会立刻把每个节点机的基本信息显示出来，如图 4 所示.

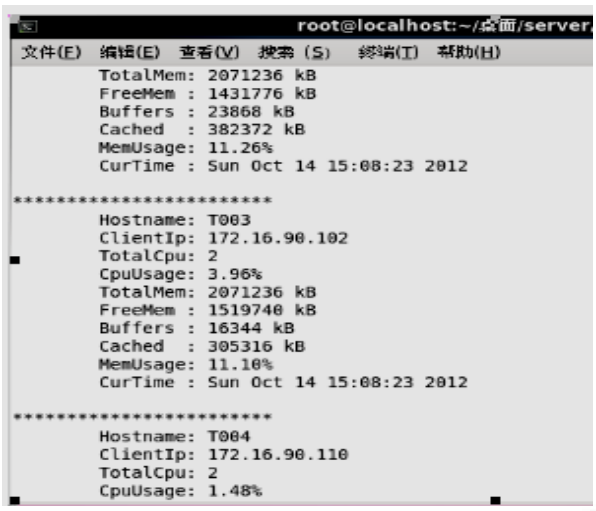

图 4 信息管理器显示界面

同时为了更好地发挥监控器的效果，我们设计了日志管理分析功能，可以在配置文件中设置开关，来决定是否开启该功能. 图 5 显示了管理节点的信息管理器的部分日志内容.

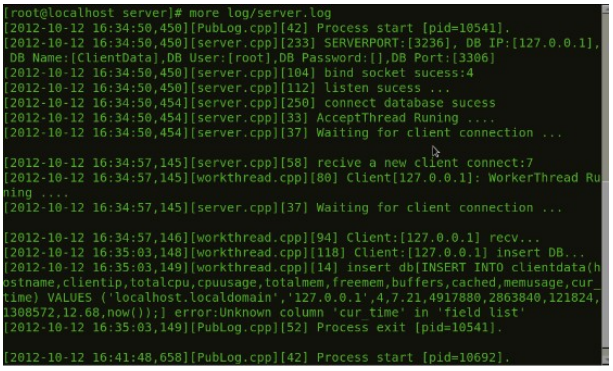

图 5 信息管理器日志文件

6 结论

监控系统的设计目的在于将简化繁琐的集群监控管理，其最终目标是通过 B/S 结构为监控人员提供友好的图形界面，方便系统管理员的远程监控. 其主要优点在于[6]:

(1) 可扩展性好. 当集群中增加一台节点服务器时,无须对修改监控系统核心程序.

(2) 通过 Linux 系统底层 Socket 编程技术传输节点信息,可以减少了监控系统对集群的不利影响.

(3) 通过/proc 虚拟文件系统采集数据，是高效、快速执行系统监控的一种方法.

(4) 节点信息发布灵活. Mysql 数据库中存储着节点信息，可以采用多种技术设计 Web 服务，从 Mysql 数据库中提取节点信息给系统监控人员.

本系统还存在一些不足之处,如缺少系统故障报警机制；缺少错误处理机制等. 这些留待下一步的研究工作中解决.

参考文献

1 百度百科.云计算.http://baike.baidu.com/view/1316082.htm.

2 王鹏,吕爽,等.并行计算应用及实战.北京:机械工业出版社,2008.86–87.

3 刘杨,肖侬,沈立.Xen 虚拟集群监控器的设计与实现.武汉理工大学学报,2010,32(20):184–188.

4 Brian Goetz.Java theory and practice:Thread pools and work queues.http://www.ibm.com/developerworks/library/j-jtp0730/index.html.

5 Xiong Q, Rong QS. The design of serial server based on semaphore under Linux. 2010 2nd IEEE International Conference on Information Management and Engineering,1: 398–400.

6 李胜利,邱昊,邵志远.一种基于 Web 的流媒体集群监控系统.计算机工程与科学,2008,30(2):5–8.