基于 Linux 集群的 Web 服务的研究和构建

程 洪 钱乐秋 洪 圆
(复旦大学计算机与信息技术系,上海 200433)
E-mail:alongwaych@yahoo.com.cn

摘 要 集群是指将两台或更多的互连的计算机整合在一起,对外表现为一个统一的计算资源。它通常有三个特征:高性能、高可用性和可伸缩性。该文在深入研究集群技术后,根据用户构建高性能、高可用性实用 Web 服务的需求,提出了一个基于 Linux 集群的解决方案,并在增强其可管理性,可靠性方面进行了探讨,使其达到实际应用需求。

关键词 集群 Linux 虚拟服务器 负载均衡 高可靠性 可伸缩性

文章编号 1002-8331-(2004)34-0158-04 文献标识码 A 中图分类号 TP393

Research and Design of Linux-based Web Server

Cheng Hong Qian Leqiu Hong Yuan
(Dept.of Computer and Information Technology,Fudan Univ.,Shanghai 200433)

Abstract: A cluster is a set of interconnected computers serving a specific need.Clustering can address three issues: high performance,high availability and scalability.After studying the clustering technology and analyzing the customer's requirement of construction of high performance,high availability and applied Web Server,we introduce a Linux-based cluster method.In addition,we do a great deal to enhance the manageability and availability of the system,so that we can do better in the aspect of meeting the applicable requirement.

Keywords: cluster,Linux Virtual Server (LVS),load balance,high availability,scalability

1 引言

当今计算机技术已进入以网络为中心的时代,Internet 用户和 Internet 流量正以惊人的速度增长,这对网络和服务器性能提出了巨大的挑战;与此同时,Web 服务中越来越多地使用 JSP、ASP、PHP 等动态主页技术,这也对服务器的性能提出了更高的要求。总而言之,未来的网络服务需要能提供更丰富的内容、更好的交互性、更高的安全性并且能承受更高的访问量,这就需要网络服务具有更高的性能、更大可用性、更好可伸缩性和更合理的价格有效性。集群技术可以很好地解决这个问题。

集群是指将两台或更多的互连的计算机整合在一起,对外表现为具有高可用性、高性能和易管理性的单一的、统一的计算资源。通过高性能网络或局域网互联的服务器集群正成为实现高可伸缩的、高可用网络服务的有效结构。该文探讨了如何利用集群技术解决一个实际应用问题,并在增强集群的可管理性和可靠性方面做出了很多努力和探讨,使集群技术符合具体的应用需求。

2 Linux 虚拟服务器集群技术研究

2.1 什么是 Linux 虚拟服务器 (LVS)

LVS 实际上是一种 Linux 操作系统上基于 IP 层的负载均衡调度技术,它在操作系统核心层上,将来自 IP 层的 TCP/UDP 请求均衡地转移到不同的服务器,从而将一组服务器构成一个高性能、高可用的虚拟服务器。

2.2 LVS 体系结构

LVS 一般由两部分组成:真实服务器和调度器。由真正运行网络服务的机器充当真实服务器的角色,提供网络服务的真实服务器的结点数目是可变的。当整个系统收到的负载超过目前所有结点的处理能力时,可以再增加真实服务器来满足不断增长的请求负载。对大多数网络服务来说,不同请求之间不存在很强的相关性,请求可以在不同的结点上并行执行,所以整个系统的性能基本上可以随着真实服务器的数目的增加而线性增长。调度器是服务器集群系统的唯一入口点 (Single Entry Point),它采用 IP 负载均衡技术。当客户请求到达时,调度器会根据真实服务器负载情况和设定的调度算法从真实服务器中选出一个服务器,再将该请求转发到选出的服务器,并记录这个调度;当这个请求的其他报文到达,该报文也会被转发到前面选出的服务器。因为所有的操作都是在 Linux 操作系统核心空间中完成的,它的调度开销很小,所以它具有很高的吞吐率。整个服务器集群的结构对客户是透明的,客户所能看到的是单一的虚拟的服务器。图 1 显示的是 LVS 集群通用的体系结构。

2.3 LVS 主要技术

2.3.1 LVS IP 负载均衡技术

LVS 采用三种 IP 负载均衡技术,即网络地址转换、IP 隧道和直接路由。

它们的大致原理分别如下:

2.3.1.1 网络地址转换:Virtual Server via Network Address Translation (VS/NAT)

通过网络地址转换,调度器重写请求报文的目标地址,根据预设的调度算法,将请求分派给后端的真实服务器:真实服

作者简介:程洪 (978-),男,硕士研究生,研究方向:软件工程。钱乐秋,男,教授、博导,研究方向:软件工程、构件技术,软件重用技术。洪圆 (980-),男,硕士研究生,研究方向:数据库。

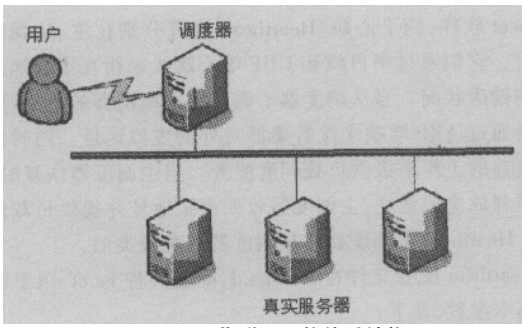

图 1　LVS 集群通用的体系结构

务器的响应报文通过调度器时,报文的源地址被重写,再返回给客户,完成整个负载调度过程。

2.3.1.2　IP 隧道 (Virtual Server via IP Tunneling (VS/TUN)

采用 NAT 技术时,由于请求和响应报文都必须经过调度器地址重写,当客户请求越来越多时,调度器的处理能力将成为瓶颈。为了解决这个问题,调度器把请求报文通过 IP 隧道转发至真实服务器,而真实服务器将响应直接返回给客户,所以调度器只处理请求报文。由于一般网络服务应答报文比请求报文大许多,采用 VS/TUN 技术后,集群系统的最大吞吐量可以提高 10 倍。

2.3.1.3　直接路由 (Virtual Server via Direct Routing (VS/DR)

VS/DR 通过改写请求报文的 MAC 地址,将请求发送到真实服务器,而真实服务器将响应直接返回给客户。同 VS/TUN 技术一样,VS/DR 技术可极大地提高集群系统的伸缩性。这种方法没有 IP 隧道的开销,对集群中的真实服务器也没有必须支持 IP 隧道协议的要求,但是要求调度器与真实服务器都有一块网卡连在同一物理网段上。

2.3.2　LVS 调度算法

负载均衡调度关心的是如何将请求流调度到各台服务器,使得各台服务器尽可能地保持负载均衡。负载调度算法设计的好坏直接决定了集群在负载均衡上的表现,设计不好的算法,会导致集群的负载失衡。一个好的负载均衡算法也并不是万能的,它一般只在某些特定的应用环境下才能发挥最大效用。因此在考察负载均衡算法的同时,也要注意算法本身的适用面,并在采取集群部署的时候根据集群自身的特点进行综合考虑,把不同的算法和技术结合起来使用。LVS 已实现了 8 种基本调度算法:轮叫调度、加权轮叫、最少链接、加权最少链接、基于局部性的最少链接、带复制的基于局部性最少链接、目标地址散列、源地址散列等。文章重点介绍一下前 4 种基本原理。

2.3.2.1　轮叫调度 (Round Robin)

调度器通过"轮叫"调度算法将外部请求按顺序轮流分配到集群中的真实服务器上,它均等地对待每一台服务器,而不管服务器上实际的连接数和系统负载。

2.3.2.2　加权轮叫 (Weighted Round Robin)

调度器通过"加权轮叫"调度算法根据真实服务器的不同处理能力来调度访问请求。这样可以保证处理能力强的服务器能处理更多的访问流量。调度器可以自动问询真实服务器的负载情况,并动态地调整其权值。

2.3.2.3　最少链接 (Least Connections)

调度器通过"最少连接"调度算法动态地将网络请求调度到已建立的链接数最少的服务器上。如果集群系统的真实服务器具有相近的系统性能,采用"最小连接"调度算法可以较好地均衡负载。

2.3.2.4　加权最少链接 (Weighted Least Connections)

在集群系统中的服务器性能差异较大的情况下,调度器采用"加权最少链接"调度算法优化负载均衡性能,具有较高权值的服务器将承受较大比例的活动连接负载。

3　基于 Linux 集群的 Web 服务的构建

3.1　实际应用需求

实际项目是成人高考网上报名系统,采用流行的 B/S 构架和 JSP 技术。客户对系统提出的要求可归结为如下 4 点:

(1)高性能要求:由于系统需要在短期内处理大量学生的网上登陆报名工作,如何及时处理用户的请求,这对 Web 服务系统的性能是一个严峻的考验。

(2)高可靠性要求:在系统开放期间,要求系统能提供每天 24 小时、每星期 7 天的不间断服务,这对 Web 服务系统的可靠性也提出了很高的要求。

(3)价格合理性要求:整个系统的构建是经济的,低成本的,能尽量利用现有硬件设备。

(4)较大可伸缩性要求:考虑到在未来的几年里,会有更多学生采用网上报名的方式,系统的访问量可增长数十倍,因此整个系统应易于扩展,以便满足未来的需求。

3.2　解决方案

在对当前的服务器集群技术做出研究并仔细分析了用户需求后,发现如果将当前的服务器集群技术应用于该项目中,则可很好地满足用户提出的需求。在该文项目中引入服务器集群技术将会产生下列优点:

(1)具有很高的性能:网络服务的工作负载通常是大量相互独立的任务,若采用集群技术,通过一组服务器分而治之,可以获得很高的整体性能。

(2)具有高的性能/价格比:集群就是将普通 PC 机、服务器通过网络设备连接起来,来提供统一的服务,与性能相当的单一服务器相比价格通常会便宜很多。

(3)具有好的可伸缩性:集群具有很好的可伸缩性,只需少量的工作就可方便地向集群增加或删除工作节点。当现有集群不能满足应用的要求时,可向集群增加新的服务器来扩充集群的处理能力。若系统采用了集群技术,则该系统中的结点数目可以增长到几千个,乃至上万个,其伸缩性远远超过单台超级计算机。

(4)具有高可用性:系统采用了集群技术,在硬件和软件上都有冗余,通过检测软硬件的故障,将故障屏蔽,由存活结点提供服务,这样可实现高可用性。

当前集群技术数不胜数,文章对多种方案进行了调查和比较。综合考虑成本,简单性,笔者研究了章文嵩博士成立的 Linux Virtual Server 的自由软件项目,发现基于 Linux 虚拟服务器 (LVS) 的集群体系结构能提供一个比较好的解决方案。Linux 虚拟服务器 (LVS) 集群通过在 Linux 内核中采用基于 IP 层的内容请求分发的负载平衡调度解决方法,将一组服务器构成一个实现可伸缩的、高可用、高性能的网络服务虚拟服务器。在 Linux 虚拟服务器的集群构架体系的基础上,通过增强系统可管理性,解决 Web 服务程序文件数据一致性等问题,充分挖掘已有机器设备的潜力,在给定的预算下获得符合应用要求的高性能、高可用性、实用低成本 Web 服务。

3.3 实现过程

3.3.1 系统体系结构

系统充分利用现有硬件设备，采用一台 P3 1.3G、256M 内存 PC 机作为主调度器，一台 P3 900 256M 内存 PC 机作为从调度器，两台 Dell PowerEdge 1750 P42.4G、512M 内存服务器作为运行 Web 服务的真实服务器，一台 Dell PowerEdge 400SC P42.4G、1G 内存 40G 硬盘服务器作为数据库服务器和 NFS 服务器。

结合实际应用的需求，LVS 的基于网络地址转换（NAT）的负载均衡技术方式由于其构建简单，对真实服务器配置少、要求低、基于私有网络安全性高，成为笔者的选择。由于要尽量利用原有硬件设备，真实服务的硬件配置差异大，所以经过分析比较，上述的 8 种基本调度算法中加权最少链接最符合实际应用需求。所设计集群系统体系结构如下：

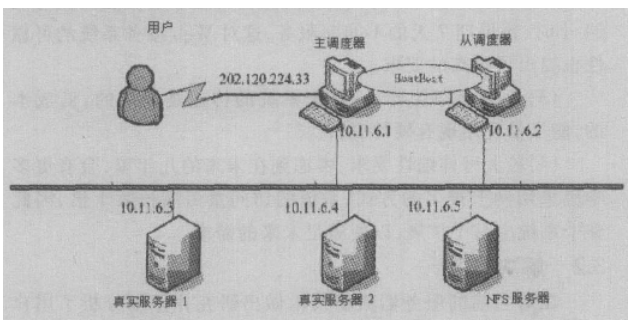

图 2 集群系统体系结构（实际系统的）

3.3.2 软件平台

操作系统均为 Redhat9.0，涉及到的软件包：Linux 2.4.20 内核、LVS 的内核补丁、Linux-2.4.20-ipvs-1.0.9.patch、Hearbeat 软件包、Ldirectord 软件包、WingMon 监控程序包、Web 服务器软件 Apache2.0 和 Tomcat4.18 等。

表 1 服务器软件配置表

服务器类别	涉及的软件包
负载均衡器	Linux 2.4.20 内核包、LVS 的内核补丁 Linux-2.4.20-ipvs-1.0.9.patch、Hearbeat 软件包、Ldirectord 软件包、WingMon 监控程序包
真实服务器软件配置	WingMon 监控程序包、Apache2.0、Tomcat4.18

3.3.3 LVS 的配置

调度器上，系统采用 LVS 提供的 ipvsadmin 设置 ip 包的转发模式和调度算法。该集群采用 NAT 模式下加权轮叫方式。配置如下：

```
ipvsadm -A -t 202.120.224.33.33:80 -s wlr
ipvsadm -a -t 202.120.224.33:80 -r 10.11.6.3:80 -m -w 2
ipvsadm -a -t 202.120.224.33:80 -r 10.11.6.4:80 -m -w 2
```

3.3.4 可靠性分析

通过 Heartbeat 和 Ldirectord 的完美组合极大地提高了系统的可靠性。

3.3.4.1 可靠的调度器

系统通过调度器将不同的用户请求调度到真实的 Web 服务器上去执行，因此调度器有可能成为系统的单一失效点。为了避免调度器失效而导致整个系统不能工作，需要设立一个从调度器作为主调度器的备份。在主调度器和从调度器上，安装 Heartbeat 软件，两个心跳（Heartbeat）进程分别在主、从调度器上运行，它们通过串口线和 UDP 等心跳线来相互定时地汇报各自的健康状况。当从调度器不能听到主调度器的心跳时，从调度器通过 ARP 欺骗来接管集群对外的虚拟地址，同时接管主调度器的工作来提供负载调度服务。当主调度器恢复时，从调度器释放虚拟地址，主调度器收回虚拟地址并提供负载调度服务。Heartbeat 主调度器和从调度器的配置类似。

Heartbeat 配置文件在/etc/ha.d，配置文件 ha.cf 用于设置一些基本参数，如下：

```
logfacility local0
keepalive 2
deadtime 10
warntime 10
initdead 10
nice_failback on
mcast eth0 225.0.0.7 694 1 1
node PriLd # 主调度器 PriLd
node BakLd # 备份调度器 BakLd
haresources :设置管理的资源
PriLd 202.120.224.33/24/eth0
```

3.3.4.2 可靠的 Web 服务

集群运作时，应当监视集群中所有真实服务器 Web 服务的运行情况并对其中的变化作出反应。如果发现真实服务器 Web 服务突然无法可用了，就需要将其从集群队列中删除，等恢复后再重新加入。当新的服务请求到来时，调度器将它负载均衡到现有的可用 Web 服务上。Ldirectord 就是这样一个服务监控程序，它监控各个真实服务器 Web 服务运行状况，并根据它修改 LVS 核心表。通过合理的配置，用它来监控运行在真实服务器上的 Web 服务，以保证对外提供可靠的虚拟统一的 Web 服务。

Ldirectord 配置文件 ldirectord.cf，基本配置如下：

```
# Global Directives
checktimeout=10
checkinterval=2
autoreload=no
logfile="local0"
quiescent=yes
# Virtual Server for HTTP
    virtual=202.120.224.33.33:80
    fallback=127.0.0.1:80
    real=10.11.6.3:80 masq
    real=10.11.6.4:80 masq
    service=http
    request="jsp/wingsoft/index.jsp"
    receive="Test Page"
    scheduler=rr
    protocol=tcp
    checktype=negotiate
```

3.3.5 数据同步的实现

该系统是一个提供 Web 服务的集群系统，每台服务器的 Web 资料必须一致，如果对 Web 资料的内容进行更新、增加或删除，那么如何使所有服务器之间数据同步呢？为了保证数据

同步问题，这里使用网络文件系统 NFS。一般而言，NFS/CIFS 服务器可支持到 6 个繁忙的服务器结点，对于应用而言已经足够了。NFS 使用了一组活动的进程，为整个集群提供了一个统一存放 Web 应用程序的目录，这样方便了更新和管理。上传好 Web 应用程序到目录/usr/local/webS 后，在文件/etc/exports 中添加一行 /usr/loca/webs (rw)，这表明这个目录可以在装载 NFS 的任何远程系统上使用，并有读写权限。在每个真实服务器上，在其文件/etc/fstab 中添加下面一行：

10.11.6.5 :/usr/local/webs /usr/local/tomcat/webapps/ROOT/wingsoft，

将 webs 映射为本地目录/usr/local/tomcat/webapps/ROOT/wingsoft。

3.3.6 性能监控及管理

为了获得集群的运行情况，便于对集群进行统一管理，需要对各个真实服务器进行性能监控并能对它们进行统一有效的控制。WingMon 是笔者自主开发的基于主从体系结构的一套监控管理系统。它由三部分组成：运行在各真实 web 服务器的统计管理程序 WingmonClient、运行在调度器节点的中央监控程序 WingmonServ 和运行在调度器节点的用于向各个真实 Web 服务器发送命令的 WingCmd。

3.3.6.1 Wingclient

Linux 提供了一个进入内核的窗口即/proc 的伪文件，你们可以象阅读普通文件一样阅读它，获取所感兴趣的数据。Wingclient 启动时，fork 出一个子进程，每隔一段时间，这个子进程阅读/proc，收集有关真实 web 服务器网络，CPU 和内存利用率的信息通过 UDP 数据报发送给中央监控程序；WingClient 父进程监听固定端口，接受从调度器发送的控制命令如重起 Web 服务，操作系统等。

3.3.6.2 WingmonServ

运行在调度器上，中央监控程序 WingmonServ 监听各个真实服务器 WingClient 程序发送过来的数据报，绘制出接近实时性能的图。

3.3.6.3 Wingcmd

通常运行在调度器上，通过 UDP 包，广播对各个真实服务器的控制命令，如重起操作系统等。

以下是统计管理程序和中央监控程序通信数据报的结构：

```
Struct cprStats{
  Int cp ;//CPU 利用率
  Int rm ;//内存利用率
  Int ps ;//接受到的数据报
  Int pr ;// 发送的数据报
  Int dr ;//读硬盘数
  Int dw ;//写硬盘数
}
```

由于运行在真实服务器的统计管理程序每隔很长的时间（通常为一秒）才会收集一次真实服务器的负载信息传递给中央监控程序，所以它对真实服务器的性能和整个集群的网络负载造成的影响几乎可以忽略。

WingMon 软件包中，运行在在调度器上的 WingmonServ 和 Wingcmd 可以以命令行方式启动，为了使运行在真实服务器上管理统计程序 WingMonClient 能自启动，在真实服务器/etc/rc.d/rc3.d/目录增加一个指向 WingMonClient 的软连接即可。

4 结论

集群技术方案很多，笔者采用的是开源的基于 Linux 的 LVS 方案，充分发掘了已有硬件设备的潜力，有很大的性能价格比优势。在此基础上，采用多种方法，增强其可靠性，可管理性，大大提高了实用性，达到实际应用需求。但此方案仅适用中小规模的 Web 服务应用要求，如其采用的 VS/NAT 技术，NFS 技术大大限制了其更高的扩展性，要构建更高性能，更高可靠性的，适应大规模 Web 服务应用的集群还有很多需改进的地方。（收稿日期：2004 年 6 月）

参考文献

1.Simon Horman.LVS Tutorial.www.ultramonkey.org
2.章文嵩.Linux 服务器集群系统（LVS）.www.ibm.com/cn
3.Joseph Mack.LVS-HOWTO.www.linuxvirtualserver.org
4.Alex Vrenios，sams.Linux Cluster Architecture.2002

（上接 123 页）

表明，无论主系统、子系统如何操作（以串口通讯为例），均可靠工作。

5 结论

该文提出的任务型软件“看门狗”，借鉴硬件设计中的“看门狗”技术，利用单片机的定时中断，只需少量资源，就可有效地提高 MCU 执行时间的效率，提高单片机软件系统的抗干扰和容错、纠错能力；且设计方案独立于任务处理模块之外，正常情况下，不干扰、不影响任务的运行状态和结果，切实可行。在福建航道局航标灯监控系统的应用表明：这种设计有效地提高了系统的稳定性、可靠性，具有一定的应用和推广价值，并可以借鉴应用在其它嵌入式系统软件开发中。（收稿日期：2004 年 5 月）

参考文献

1.何立民.MCS-51 系列单片机应用系统设计[M].北京：北京航空航天大学出版社，1990
2.周航慈.单片机程序设计技术[M].北京：北京航空航天大学出版社，1997
3.刘先昆，张琴，张骏等.混凝土搅拌站计算机控制系统中的“看门狗”设计
4.李伯成，柳宝堂.“看门狗”及其应用[J].遥测遥控
5.薛天宇.一种实时多任务系统软件设计方法[J].电子技术应用，2001；(4)：11~15
6.梁景新，何晨，诸鸿文.小型静态实时多任务架构[J].系统工程与电子技术，2001；(7)：69~71
7.张秋菊，王凤贺.多任务调度算法在单片机控制系统中的应用[J].光电对抗与无源扰，2002；(5)：23~25
8.李雅梅，杨顺，李新春.单片微机多任务处理能力分析[J].辽宁工程技术大学学报（自然科学版），2002；(5)：342~343
9.蒋翔.单片机程序设计中运用事件驱动机制[J].电子技术应用，2002；(7)：28~30