多网卡 bonding 技术在无线广告系统中的应用①

Application of Multiple Network Channels Bonding in Wireless Multimedia Advertisement System

贺　剑　黄　仁　(重庆大学 计算机学院 重庆 400030)

摘　要： 简单地介绍了多网卡链路绑定策略，重点讨论了 Linux 下的 bonding 技术及其基于 bonding 技术的几种负载均衡算法。给出了无线多媒体广告系统的主要架构，分析了负载均衡算法的差异性及适用情况，给出了 MAC 地址异或算法在该系统中的应用实现，并做了相关测试，测试结果表明该方案成功地解决了本系统的网络瓶颈。

关键词： Linux Bonding MAC 地址异或算法 无线多媒体广告 分布式系统

1　引言

随着基于网络应用的复杂性及重要性不断增加，网络服务器承担的压力也愈来愈重。如果单靠升级网络硬件(如千兆网卡及千兆交换机)来提高网络性能会对网络服务器的高性价比带来负面影响。因此，在利用现有网络硬件设备的环境下，通过软件方式提高网络吞吐率已经成为网络服务器亟待解决的问题。多网卡链路绑定策略 MNCB(Multiple Network Channel Bonding)[1]正是在这种背景下提出的，它主要针对提高服务器的网络吞吐率及高可靠性。

目前 Linux 系统的 bonding 技术[2]、3com 公司的 DynamicAccess 技术、Intel 公司开发的 AFT (Adapter Fault Technology，网络容错/网络连接备份)和 ALB(Adaptive Load Balancing，自适应负载平衡)等技术都在研究将多个网卡接口 bonding 在一起的链路聚集(link aggregation 或 trunking)技术。IEEE(美国电气与电子工程师学会)已经把局域网链路聚集技术提升为行业标准，即所谓的 IEEE802.3ad 协议。3Com Dynamic Access 美中不足的是对非 3Com 服务器网卡的其他网卡只能绑定 2 块，而 3Com 自己的服务器网卡则在一个网卡组中最多可绑定 8 块。Intel AFT 主要依靠备份网络链接提高网络可靠性，ALB 可以将多网卡虚拟成一块网卡，但是 ALB 技术主要是通过预先静态设置网络负载分配，容易由于分配不合理造成网络负载不均衡。Linux 下的技术都具有开源性质，根据需求可以选择合理的负载均衡算法，甚至可以根据具体适用环境得情况修改传统算法提出符合实际需求的负载均衡算法，所以 bonding 技术也被广泛应用。

1　Linux的bonding技术的原理及负载均衡算法

1.1 Linux 的 bonding 技术原理

Linux 的 bonding 技术是网卡驱动程序之上、数据链路层之下实现的一个虚拟层，通过这种技术，服务器接在交换机上的多块网卡不仅被绑定为一个 IP，MAC 地址也被设定为同一个，进而构成一个虚拟的网卡，工作站向服务器请求数据，服务器上的网卡接到请求后，网卡根据某种算法智能决定由谁来处理数据的传输。

它的工作机制[3]是由 bonding 驱动程序统一管理和配置 bonding 设备内的网卡资源。在发送数据时，对于数据链路层来说，网卡设备是透明的。应用程序发送的数据包经由 IP 层和数据链路层发往 bonding 设备，而不是那些具体的物理接口。bonding 设备驱动根据事先设定好的传输模式(算法)调度设备中的网卡资源，由网卡把数据发送出去。同时，bonding 设备驱动还可以控制设备的其它基本属性，如设定设备的 IP 地址、MAC 地址，打开或者关闭设备等。

① 收稿时间:2008-12-15

1.2 Linux 的 bonding 技术的负载均衡发送算法

多网卡 bonding 设备的主要功能是通过预先指定的数据传输算法对设备内的网卡设备统一调配，以提高网络的吞吐量。因此，数据传输算法在多网卡 bonding 设备驱动中处于核心地位。目前 Linux 的发送算法最主要的有三种：轮转算法(Round-Robin)[4]、备份算法(Active-Backup)、MAC 地址异或算法(MAC-XOR)[5]。下面对目前这三种主要算法进行简单分析。

1.2.1 轮转算法

该算法是基于公平原则进行的，它将所有相同优先级的网卡设备维持在一个循环队列(slave 设备链表中)，bonding 设备驱动在这些网卡设备中顺序轮流选择。

1.2.2 备份算法

该算法属于非负载均衡算法，它将多个网卡接口中的一个接口设定为活动状态，其他的接口处于备用状态。当活动接口或者活动链路出现故障时，启动备用链路，进行工作。

1.2.3 MAC 地址异或算法

异或算法又称为散列算法。它的算法思想是由服务器的 MAC 地址和客户端的 MAC 地址共同决定每个数据包的发送接口号。由服务器的 MAC 地址和客户端的 MAC 地址共同决定每个数据包的发送端口号，由源 MAC 地址和目的 MAC 地址进行异或计算，并将异或结果对接口数求余计算。由于发送到同一个客户端的数据流经过同一个链路，因此数据包能够有序到达客户端。

2 Linux的bonging技术的应用

2.1 项目背景

随着楼宇视频广告这种新兴媒体的日益成熟，其生动的表现形式、分众的清晰定位、强制的收视效果备受中高端广告主的青睐，成为中国新传媒市场的主流广告方式之一。当前楼宇视频广告也存在一些不足，例如：需要人工更换视频广告、人工维护和定期检测视频终端机等。

该管理系统的建立，就是为了达到对楼宇广告的广告视频、楼宇终端机和客户等信息的一体化管理，改变当前信息管理混乱的局面。

2.2 无线广告系统的系统架构

根据楼宇视频广告的特点以及对网络分布结构的需求，本系统采用了 C/S 和 B/S 混合的异构软件体系架构，如图所示，主要分为一下几个主要组成部分：

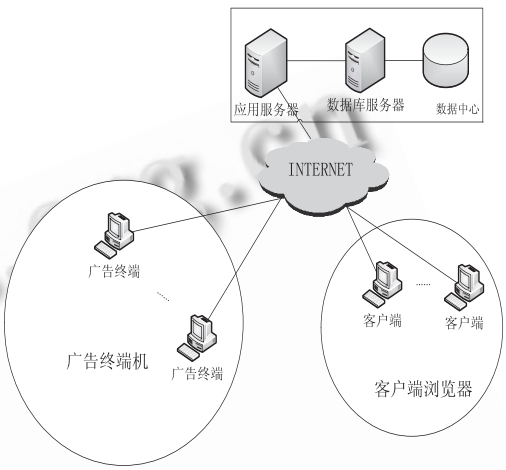

图 1 系统结构图

2.2.1 应用服务器及数据库服务器

应用服务器主要运行管理系统的服务器端，主要有以下功能：Web Server 接收和处理来自客户端的业务请求，例如：订制广告视频播放列表、添加广告视频文件等；Application Server 监控和管理广告视频终端机信息，例如：维护广告视频终端机在线列表，监控终端机的运行状态；Ftp Server，基于无线模块向终端机传输视频、图片、文字、播放列表等文件信息。服务器端采用多线程技术实现以上模块功能，以及模块之间的通信和交互，数据库服务器采用了 Mysql5.2 为应用程序提供了数据存储及管理功能。

2.2.2 广告终端机

终端机运行管理系统的终端机客户端，与服务器端形成 B/S 的架构模式，主要有以下功能：Application Client，与服务器端进行通信和交互；Ftp Client，接收视频、图片、文字、播放列表等文件信息。Player UI 根据终端机上的 XML 格式的播放列表，进行视频、图片、文字等广告信息的播放，并向 Application Client 报告自己的状态。

2.2.3 浏览器客户端

该系统和用户的交互主要通过浏览器客户端的方式进行。例如：用户可以通过浏览器客户端，制定播放列表、向广告终端机发送命令和传输播放列表、查看在线广告终端机的状态。

2.2.4 Linux Bond 发送算法的比较和选择

当服务器端向终端机传输视频文件时，通常都是对某一片区或者某一楼宇的所有终端机同时发送，服务器的网络带宽瓶颈就显现出来了。为了解决这一问题，下面讨论一下目前存在的三种 bonding 算法哪种更适合：

轮转算法的负载均衡性能最好，资源利用率也很高。但是我们知道如果一个连接或者会话的数据包从不同的接口发出的话，中途再经过不同的链路，在客户端很有可能会出现数据包无序到达的 问题 ，而无序到达的数据包需要重新要求被发送，这样网络的吞吐量就会下降。在视频文件传送的过程中，难免存在某一视频文件的数据包无序到达的情况，只能请求重传，这大大地降低了网络的传输能力。

备份算法提高了网络接口的稳定性和可用性，不过本系统期待大幅度提高网络的吞吐能力，而该算法资源利用率低，网络负载不均衡，无法提高网络吞吐能力，所以该算法也不太适合本系统。

MAC 地址异或算法保证了相同客户端的数据包从同一网卡发出去，避免了无序到达的情况，同时多块网卡同时运行，达到了提高网络负载的目的。当然，该算法也有不足之处，在只有一个客户机访问服务器时，存在资源的利用率低，负载不均衡的情况。但是，本系统需要解决的是服务器同时向多个终端机传输文件时导致的网络性能瓶颈，所以 MAC 地址异或算法比较契合本系统的需求。

2.3 Linux bonding 的 MAC 异或地址算法的实现

如图 2 所示，异或算法的实现很简单，bond_set_mode_ops()函数在初始化数据传输函数时，选择异或模式，让 bond_set_mode_ops()函数就会使 hard_start_xmit 函数指针指向异或模式的数据传输函数 bond_xmit_xor()。

bond_xmit_xor()的入口参数是 struct sk_buff *skb 和 struct net_device *dev。其中，skb 结构用来存储从数据链路层传输过来的数据，而 dev 结构是用来存放用来发送的网卡设备。

bond_xmit_xor()首先把客户端的 MAC 地址 h_dest 与 bonding 设备的 MAC 地址 dev_addr 进行异或，然后对网卡设备数 slave_cnt 取余操作，得到网卡设备的编号 slave_no；其次根据网卡的设备编号，遍历整个 slave 设备链表，选择网卡设备；最后调用网卡设备的 bond_dev_queue_xmit()函数发送数据。

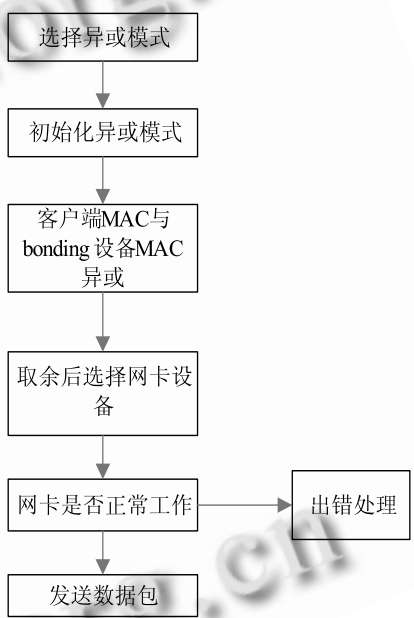

图 2　异或算法流程图

3　测试结果

截至本文投稿，Linux bonding 技术已经被应用到该系统中，试用阶段情况较好。在此给出系统开发初期对基于 MAC 地址异或发送算法的性能的简单实验。测试硬件环境为一台服务器，两台客户端，测试软件为 Netperf。Netperf 是一种网络性能的测量工具，主要针对基于 TCP 或 UDP 的传输。Netperf 根据应用的不同，可以进行不同模式的网络性能测试，即批量数据传输(bulk data transfer)模式和请求/应答(request/reponse)模式。Netperf 测试结果所反映的是一个系统能够以多快的速度向另外一个系统发送数据，以及另外一个系统能够以多块的速度接收数据。

按以下步骤进行测试，首先服务器用单个百兆网

卡，两台客户端同时请求服务器，服务器向客户端同时传输视频文件，待传输速率稳定后，获取试验数据；之后服务器采用两个百兆网卡，bonding 设备之后，两台客户端再次请求服务器。实验数据(稳定传输后的平均数据，忽略了选择网卡，链路建立阶段的开销)如下：

表 1 网络测试数据表

	单网卡	双网卡
延时(us)	42	45
吞吐率(Mb/s)	90	175.49

根据以上实验结果得知，多网卡 bonding 技术对于提高网络吞吐率有着非常好的效果。实验中，服务器采用双网卡正好对应两台客户端，负载均衡好，所以网络吞吐率的提高幅度非常大，在实际应用中，基于 MAC 地址异或发送算法存在网络负载不均衡，吞吐率性能提高幅度会相应有所下降，但是已经能够解决在本系统中存在的网络瓶颈问题。

4 结论

本文中的基于多网卡 bonding 技术的无线广告系统采用了基于 B/S 和 C/S 混合的异构软件体构，实现了对信息的管理以及硬件设备的控制。多网卡链路绑定的方案在不过多增加本系统的开发成本基础上，较好地解决了服务器网络性能的瓶颈问题。但是另一方面，由于基于人工管理广告机模式的运营方式已经存在和普及，因而本系统的推广有一定阻力，然而随着市场需求的扩大及高标准，本系统将体现出它的优势。

参考文献

1 Gray R, Wright W, Stevens R. TCP/IP 详解.陆雪莹,蒋惠等译.北京:机械工业出版社, 2002.

2 Forouzan BA, Fegan SC. TCP/IP protocol suite.北京:清华大学出版社, 2004.

3 毛德操,胡希明.LINUX 内核源代码情景分析.杭州:浙江大学出版社, 2006.

4 路明怀,龚正虎.Linux 服务器下多网卡负载均衡的研究与实现.计算机与信息技术, 2006,(6):24.

5 A Cost-effective Approach to Improve Server Performance and Fault Tolerance.http://itpapers. zdnet.com/