



(12) 发明专利

(10) 授权公告号 CN 101977162 B

(45) 授权公告日 2012. 01. 18

(21) 申请号 201010572227. 5

审查员 胡鑫

(22) 申请日 2010. 12. 03

(73) 专利权人 电子科技大学

地址 611731 四川省成都市高新区(西区)西源大道 2006 号

(72) 发明人 张超 杨瀚程 李庆 刘晓斌 朱姝 张小倩

(74) 专利代理机构 电子科技大学专利中心 51203

代理人 周永宏

(51) Int. Cl.

H04L 12/56 (2006. 01)

(56) 对比文件

US 2005/0078601 A1, 2005. 04. 14, 全文.

CN 1937557 A, 2007. 03. 28, 全文.

CN 101753610 A, 2010. 06. 23, 全文.

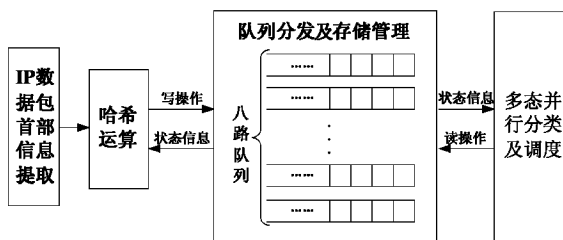
权利要求书 1 页 说明书 4 页 附图 2 页

(54) 发明名称

一种高速网络的负载均衡方法

(57) 摘要

本发明公开了一种高速网络的负载均衡方法。本发明针对现有的负载均衡方法均衡度不高的问题,提出了一种高速网络的负载均衡方法,包括如下步骤:IP 数据包首部信息提取,哈希运算,队列分发及存储管理,多态并行分类及调度。本发明的负载均衡方法利用了静态均衡与动态自适应均衡相结合的动态自适应哈希均衡策略,使得本方法可以有效利用当前流量负载情况,进而使得高速数据流可以均衡分配到多个低速网络接口上,提高各个子通道相互间的均衡度。



1. 一种高速网络的负载均衡方法,包括如下步骤:

S1. IP 数据包首部信息提取,包括提取源 IP 地址、目的 IP 地址、源端口号、目的端口号;

S2. 哈希运算,对提取的 IP 首部信息进行哈希运算,将哈希运算的结果对 4 求模取余数,根据求模得出的值来分配 IP 数据包的指示信息到相应的存储队列中;

S3. 队列分发及存储管理,包括队列的写操作管理和读操作管理;

S4. 多态并行分类及调度,该过程根据各个队列反馈的空、满状态信息和端口流量指示信息,通过控制两次轮询操作之间的时间间隔,采用最小流量优先和满队列优先的策略轮询各个队列,从各个队列中轮询读出相应的 IP 分组的指示信息,根据指示信息从数据缓存器中读出相应的数据分组,转发到相应的目的端口上;

具体包括如下步骤:

S41. 每次转发完一个数据分组时,保存端口流量信息并询问各个队列;

S42. 首先判断是否存在满队列,如果存在,再判断是否存在多个满队列,如果只有一个满队列,就计算其相应的轮询等待时间,启动等待时间计时器,到达轮询等待时间时,转发该满队列的数据分组到相应的目的端口上;如果存在多个满队列,根据这几个满队列统计的已发送数据流量值,选中满队列中端口流量值最小的队列,计算其轮询等待时间,启动等待时间计时器,到达轮询等待时间时转发该队列的数据分组到相应的目的端口上,另外,如果多个满队列中有两个以上的队列的数据流量值相等,且它们的流量值最小,则转发端口号最小的满队列的数据分组到相应的目的端口上;如果不存在满队列,选中所有队列中的非空队列;

S43. 比较各个非空队列已发送的数据流量值,判断非空队列中是否只有一个队列的流量值最小,如果是,就计算该队列的轮询等待时间,启动等待时间计时器,到达轮询等待时间时,就直接转发该队列对应的数据分组到相应的目的端口上;否则,选中端口号最小的队列,计算其轮询等待时间,启动等待时间计时器,达到轮询等待时间时,转发该队列对应的数据分组到相应的目的端口上。

2. 根据权利要求 1 所述的方法,其特征在于,步骤 S2 中所述的哈希运算为 IPSX 哈希算法。

一种高速网络的负载均衡方法

技术领域

[0001] 本发明属于负载均衡技术领域,尤其涉及一种高速网络的负载均衡方法。

背景技术

[0002] 为了提高网络的安全性,一些安全监测设备用来对用户流量进行实时监测,可随着网络速度的不断提高,业务量和数据流量的不断增长,使单一的服务器根本无法承担,此时如果对设备进行硬件升级,将造成资源浪费且投入成本较高,针对这种情况,一些负载均衡的方法应运而生。

[0003] 负载均衡的方法是一种廉价、有效、透明的方法,可以扩展现有网络设备和服务器的带宽,增加吞吐量,加强网络数据处理能力,提高网络的灵活性和可用性。负载均衡的主要原理是:高速网络的数据流送至负载均衡器,负载均衡器根据某种方法将高速数据流均衡分配到多台低速的服务器上,实现对高速网络的流量均衡和监控。所以在负载均衡器中负载均衡方法至关重要。

[0004] 目前的负载均衡方法主要有静态均衡和动态均衡,静态均衡方法主要有:哈希均衡算法(哈希运算)、静态优先权调度算法、轮询调度算法、权重轮询调度算法,动态均衡方法主要有:最少连接数均衡算法和最快响应时间均衡算法。评价一种负载均衡方法优劣的一个重要指标就是均衡度 L ,即各个服务器间的带宽利用率偏差,其计算公式为: $L = |flow1 - flow2| / flow$,其中 $flow1$, $flow2$ 为两个通道的当前流量值, $flow$ 为两个通道的平均理想流量值。以上各种负载均衡方法在对各个服务器分配数据流量时,都没有考虑各个服务器的当前流量负载情况,所以存在各个服务器间的均衡度不高的问题。

发明内容

[0005] 本发明的目的是为了解决现有的负载均衡方法均衡度不高的问题,提出了一种高速网络的负载均衡方法。

[0006] 为了实现上述目的,本发明的技术方案是:一种高速网络的负载均衡方法,具体包括如下步骤:

[0007] S1. IP数据包首部信息提取,包括提取源IP地址、目的IP地址、源端口号、目的端口号;

[0008] S2. 哈希运算,对提取的IP首部信息进行哈希运算,将哈希运算的结果对4求模取余数,根据求模得出的值来分配IP数据包的指示信息到相应的存储队列中;

[0009] S3. 队列分发及存储管理,包括队列的写操作管理和读操作管理;

[0010] S4. 多态并行分类及调度,该过程根据各个队列反馈的空、满状态信息和端口流量指示信息,通过控制两次轮询操作之间的时间间隔,采用最小流量优先和满队列优先的策略轮询各个队列,从各个队列中轮询读出相应的IP分组的指示信息,根据指示信息从数据缓存器中读出相应的数据分组,转发到相应的目的端口上。具体包括如下步骤:

[0011] S41. 每次转发完一个数据分组时,保存端口流量信息并询问各个队列;

[0012] S42. 首先判断是否存在满队列, 如果存在, 再判断是否存在多个满队列, 如果只有一个满队列, 就计算其相应的轮询等待时间, 启动等待时间计时器, 到达轮询等待时间时, 转发该满队列的数据分组到相应的目的端口上; 如果存在多个满队列, 根据这几个满队列统计的已发送数据流量值, 选中满队列中端口流量值最小的队列, 计算其轮询等待时间, 启动等待时间计时器, 到达轮询等待时间时转发该队列的数据分组到相应的目的端口上, 另外, 如果多个满队列中有两个以上的队列的数据流量值相等, 且它们的流量值最小, 则转发端口号最小的满队列的数据分组到相应的目的端口上; 如果不存在满队列, 选中所有队列中的非空队列;

[0013] S43. 比较各个非空队列已发送的数据流量值, 判断非空队列中是否只有一个队列的流量值最小, 如果是, 就计算该队列的轮询等待时间, 启动等待时间计时器, 到达轮询等待时间时, 就直接转发该队列对应的数据分组到相应的目的端口上; 否则, 选中端口号最小的队列, 计算其轮询等待时间, 启动等待时间计时器, 达到轮询等待时间时, 转发该队列对应的数据分组到相应的目的端口上。

[0014] 步骤 S2 中, 所述的哈希运算为 IPSX 哈希算法。

[0015] 本发明的有益效果: 本发明利用将静态均衡与动态自适应均衡相结合的动态自适应哈希均衡策略, 提出了一种前端利用静态哈希均衡算法和后端利用多态并行分类及调度算法的高速网络负载均衡方法, 该方法利用了当前流量负载情况, 使得高速数据流可以均衡分配到多个低速网络接口上, 提高各个子通道相互间的均衡度。

附图说明

[0016] 图 1 是本发明的高速网络负载均衡方法的逻辑框图。

[0017] 图 2 是本发明的高速网络负载均衡方法的流程图。

[0018] 图 3 是本发明的多态并行分类及调度算法的流程图。

具体实施方式

[0019] 下面结合附图, 给出本发明的具体实施例。需要说明的是: 实施例中的参数并不影响本发明的一般性。

[0020] 以双通道 2.5G (0C-48 接口、VC-4-16C 级联技术) 的高速数据网络, 将双通道 2.5G 数据流均衡到 8 个千兆以太网 (GBE, GigaBit Ethernet) 接口上为例, 假设实现该方法的系统称为均衡器。

[0021] 本发明的具体设计思想是: 采用将静态均衡与动态自适应均衡相结合的动态自适应哈希 (HBDC, Hash Balancing and Dynamic Compensation) 均衡策略, 即是均衡器前端采用哈希均衡算法将高速数据流映射到各个队列中, 各个队列存储的是网际协议 (IP, Internet Protocol) 数据的指示信息, IP 数据部分缓存在一个数据缓存器中, 实现长期的统计流量均衡; 均衡器后端采用多态并行分类及调度算法, 根据流量最小优先和满队列优先的原则轮询各个队列, 从队列中读出 IP 数据的指示信息, 再从数据缓存器中取出相应的 IP 数据, 根据端口指示信息将 IP 数据流分配到相应的 GBE 端口上, 实现高速网络瞬时的流量均衡。

[0022] 具体如图 1 所示: 首先提取出 IP 数据包的首部信息, 对其首部信息进行哈希运算,

同时将 IP 数据存储到一个数据缓存器中,根据哈希运算结果将双通道 2.5G 高速 IP 数据的长度及其在缓存器中的存储首地址等指示信息分配到 8 个队列中,保证了报文的先后顺序和流的相关性,实现对双通道 2.5G 数据流的长期静态均衡;后端采用多态并行分类及调度算法,根据流量最小优先的原则实现对 8 个队列的轮询调度,轮询读出各个队列中的 IP 数据的指示信息,根据指示信息读出数据缓存器中对应的 IP 数据,并发送到相应的 GBE 端口上,在 GBE 端口上对 IP 数据及其指示信息进行分类重组,实现流量的瞬时动态均衡。其中每个队列对应一个 GBE 端口。根据多态并行分类及调度返回的指针信息及 8 个队列的空满状态信息控制对 8 个队列的写操作。

[0023] 下面结合图 2、图 3 对高速网络负载均衡过程作进一步说明,具体包括如下步骤:

[0024] S1. IP 数据包首部信息提取,针对每个 2.5G 高速数据通道,提取源 IP 地址、目的 IP 地址、源端口号、目的端口号;

[0025] S2. 哈希运算。对提取出 IP 数据包的首部信息进行哈希运算,将哈希运算的结果对 4 求模取余数,根据求模得出的值来分配 IP 数据包的指示信息到相应的存储队列中。

[0026] 当上述哈希运算为 IPSX 哈希算法时,所达到的性能最优。由于 IPSX 哈希算法为本领域的现有公知技术,因此在这里不再作详细描述。

[0027] S3. 队列分发及存储管理。该过程包括两个方面:队列的写操作管理和读操作管理。如图 2 所示,写操作的实施过程是:根据各个队列反馈的状态信息和哈希运算结果来产生相应队列的写使能,如:当哈希运算结果为 0 且 0 队列还未写满时就产生队列 0 的写使能,向队列 0 写入相应的 IP 数据的指示信息;同理,产生其它 7 个队列的写使能,向各个队列写入相应的 IP 数据的指示信息。读操作的实施过程主要根据后面的多态并行分类及调度来完成。

[0028] S4. 多态并行分类及调度。该过程根据各个队列反馈的空满状态信息,通过控制两次轮询操作之间的时间间隔,采用最小流量优先的策略轮询各个队列,从各个队列中轮询读出相应的 IP 分组的指示信息,根据指示信息从数据缓存器中读出相应的数据分组,转发到相应的 GBE 端口上。如图 3 所示,具体包括如下步骤:

[0029] S41. 每次转发完一个数据分组时,存储端口流量信息并询问 8 个队列;

[0030] S42. 首先判断是否存在满队列,如果存在,再判断是否存在多个满队列,如果只有一个满队列,就计算其相应的轮询等待时间,启动等待时间计时器,到达轮询等待时间时,进行相应的读操作,转发该满队列的数据分组到相应的 GBE 端口上;如果存在多个满队列,根据这几个满队列统计的已发送数据流量值,选中满队列中端口流量值最小的队列,计算其轮询等待时间,启动等待时间计时器,到达轮询等待时间时,进行相应的读操作,转发该队列的数据分组到相应的 GBE 端口上;另外,如果多个满队列中有两个以上的队列的数据流量值相等,且它们的流量值最小,进行相应的读操作,转发端口号最小的满队列的数据分组到相应的 GBE 端口上;如不存在满队列,在非空队列中选出端口流量最小的队列,进行相应的读操作,转发该队列的数据分组到相应的 GBE 端口上;

[0031] S43. 比较各个非空队列已发送的数据流量值,判断非空队列中是否只有一个队列的流量值最小,如果是,就计算该队列的轮询等待时间,启动等待时间计时器,到达轮询等待时间时,就直接转发该队列对应的数据分组到相应的 GBE 端口上;否则,选中端口号最小的队列,计算其轮询等待时间,启动等待时间计时器,达到轮询等待时间时,转发该队列

对应的数据分组到相应的 GBE 端口上。

[0032] 需要说明的是：本实施例中 8 个队列共用一个流量统计时间计时器，如图 3 所示，当开始发送某一队列中的数据分组时，就判断其流量统计时间是否超过了系统预先设定的单位时间，如果流量统计时间超过了单位时间，则记录下各个队列的流量累加器中的数据流量值，然后清空各个流量累加器，同时流量统计时间计时器清零，重新开始计时；如果流量统计时间没有超过单位时间，则累积相应队列的流量值。连续统计 8 个单位统计时间内的流量值，求平均作为单位统计时间内的各出端口的流量值 $port[i]_{flow}$ ，则各端口速率值为：

[0033] 其中 $i = 0, 1, 2, 3, 4, 5, 6, 7$ ，分别代表端口 0、端口 1、端口 2、端口 3、端口 4、端口 5、端口 6、端口 7。 $TIME$ 为设置的统计时间计数器。

[0034] 则轮询等待时间计算公式如下： $t = (u * len_s + (1 - u) * len_n) * delta$ 。

[0035] 其中， t 为多态并行分类及调度启动等待时间， $delta$ 为出端口速率， u 为平滑系数， len_s 为已发送的数据分组长度， len_n 为发送的下一数据分组长度。

[0036] 采用本发明所述的方法后，一个负载均衡器可以将双通道 2.5G 数据流均衡分配到 8 个 GBE 端口上，且使各个 GBE 端口间的长期统计均衡度和瞬时均衡度都优于 12%（12% 为均衡度指标），使负载流量更均衡。

[0037] 本领域的普通技术人员将会意识到，这里所述的实施例是为了帮助读者理解本发明的原理，应被理解为发明的保护范围并不局限于这样的特别陈述和实施例。凡是根据上述描述做出各种可能的等同替换或改变，均被认为属于本发明的权利要求的保护范围。

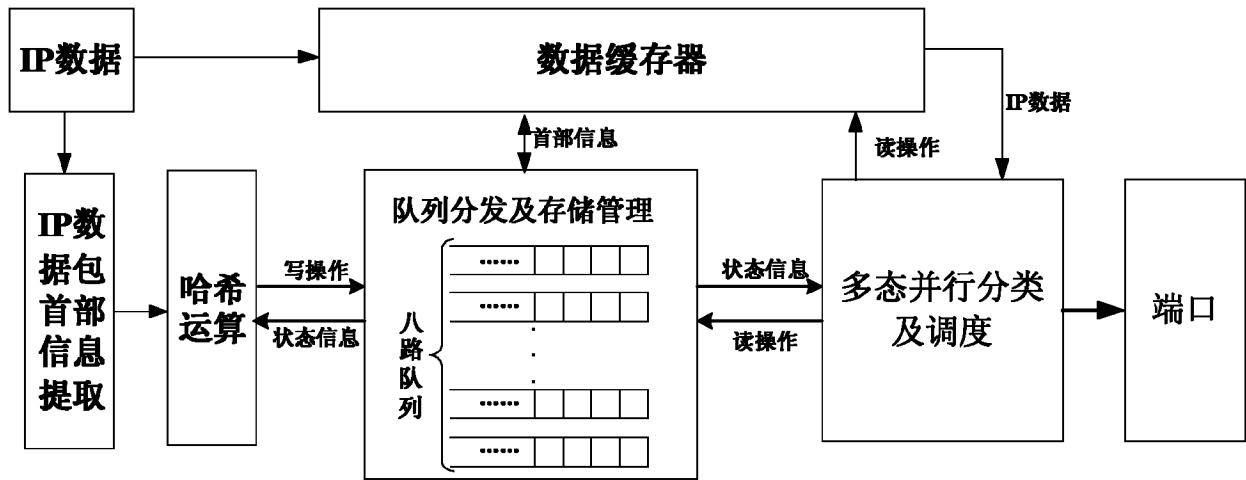


图 1

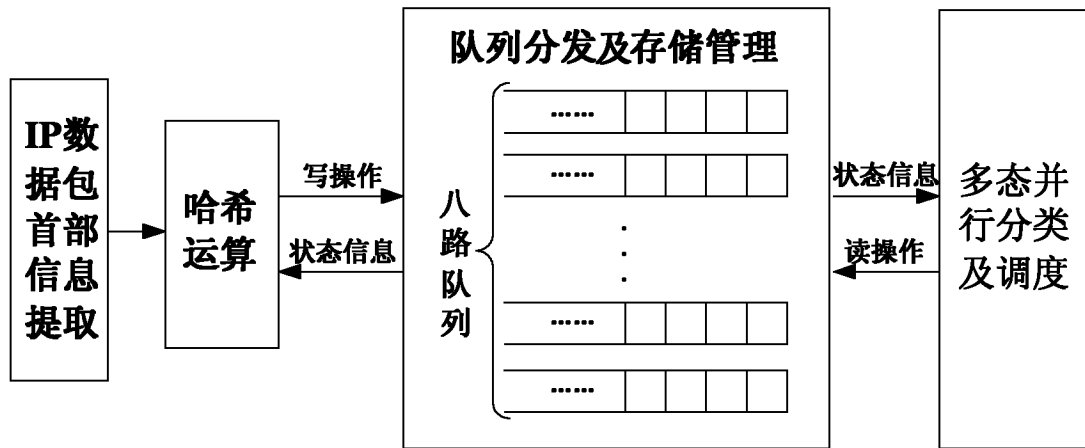


图 2

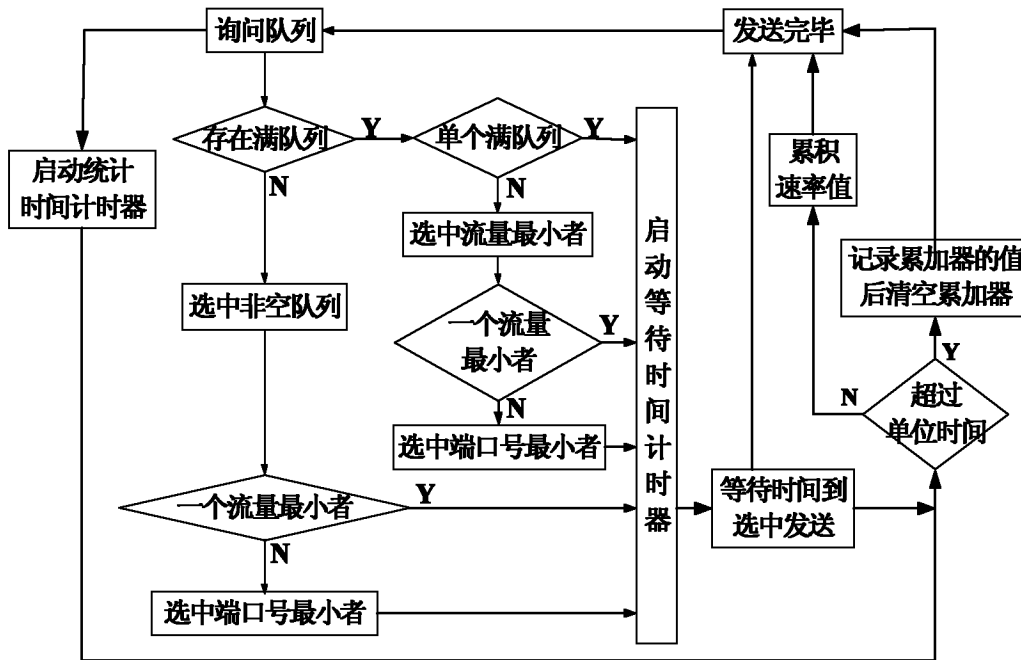


图 3