「Applied Technique」

Application of Distributed Packet Rewriting Balancing Scalable WEB Cluster Server's load

分包重写实现可伸缩WEB集群服务的负载平衡

赵岳松　周 颖

(武汉理工大学计算机专业　430070)

摘要：本文讲述低开销的DPR规范的分布式连接路由特点，及在可伸缩Web集群服务中的运用，使web集群服务具有良好的可伸缩性，容错性，负载均衡等特性。

关键词：分包重写　可伸缩　集群

1 概述

1.1 分包重写DPR（Distributed Packet Rewriting）

DPR是波士顿大学开发的可伸缩Web服务器的一个框架规范。其优点在于它的分布式连接路由协议（第4层交换），允许所有主机参与在请求重定向的过程中分布式的处理事务，因而在功能上取代了专用路由对事物的处理。

1.2 DPR的特点

(1) 透明性：允许客户区分不同的请求到集群服务的不同主机上。

(2) 强制负载分发：当一个独立节点功能不够强时，一些服务必须在一定量的节点中分离出来，请求服务是可自然分解和组成的，这样才能在节点中间最佳分发。

(3) 高可用性：在集群节点相互独立的情况下，小数量的节点失败将不会导致系统崩溃，这意味着系统在线时为了升级可以临时关闭机器，新的机器可以被加到系统中。

(4) 可伸缩性：随着集群规模的扩大系统性能将成比例的提高，尤其是系统的瓶颈问题上，系统集群规模的扩大，系统的瓶颈现象将成比例的下降。

(5) 成本效率：集群所有的能力将尽可能接近服务器所有组成元素的能力和。大大提高了性能价格比。

(6) 平滑的递降：当系统组件失败，提供的服务质量会下降，系统允许单点失败导致的服务质量的平滑下降。

(7) 灵活的连接作业：灵活的连接作业可以充分的支持资源管理功能。如允许控制负载均衡。

2 结构与策略

这个规范由共同接受和服务TCP连接的负载均衡集群主机组成，系统允许主机接收来自客户的请求，当客户试图建立TCP连接到一台主机，每个包到达时均要被检测，以决定是否要重定向到另一台主机。我们用开销小的DPR技术重定向TCP连接。每个主机的连接信息保存在哈希表和链接单中，负载的信息由集群主机定期广播来维持。

2.1 DPR的执行

在DPR的执行过程中，集群中的每台机器提供WEB服务（均具有到其他机器上的重路由能力），所有机器的IP地址被定格，允许任一机器接收请求，这些请求就可被本地服务或者重路由到另一台机器，传输到另一台机器的请求可直接被这台机器服务。DPR的执行过程（如图1），假如机器A从C接收到请求信息，用A的IP地址作为源IP，有效的重路由这请求从A到B（此时C还继续发送请求到A），则B将为C服务。我们可以用IP-IP的技术让机器B获知机器C的IP地址以便于服务。

用算法去平衡负载以加强DPR功能的分布式请求。每台机器内部获得升级表，包括集群的所有其他机器的信息，如IP地址和实时负载等信息。主机中断对其它机器广播它们的负载情况，这样可以确定是否用重路由请求，或为本地机器服务。而且，每台机器将获得路由表，含有重定向连接的信息。当机器A从C获得一个新的请求时，机器A首先检测自己的负载，如果低于一个特定的门槛值，它将服务于这个请求，否则它将在路由表上创建一个新的记录，并向当前的机器提交请求，获得最新的最低负载值，以免机器出现重创。（门槛值会依照特定的因素如CPU的速度、内存等进行调整），不同的测试负载方法中，用一定数量的开放TCP连接是获得最佳结果的方法。当每一时刻连接每台机器时，还应考虑一些其它影响负载值的因素，（如CPU的利用值，重定向TCP连接的数目，活动套接字和这些值综合一起所成的因素）

2.2 连接路由

我们来看一个客户从WEB服务中请求事件的情况：首先，客户完成了主机的域名到一个初始的IP地址。接下来，这个IP地址所在的分布式系统中的主机必须选择一个机器对其请求进行服务。有许多方法得到第一映射（从域名到初始IP地址）如：在应用中能被编码，如在Netscape浏览器Navigator访问Netscape主页一样，另一方面，这个映射可以通过单一域名服务的IP地址广告DNS来完成。同样的，有许多方法得到第二映射（从初始IP地址到真实的主机）如：可在应用级得到，可用HTTP重路由方法或用服务器的分配器得到。

当为可伸缩WEB服务器初始调度执行连接路由时，用域名服务到IP地址的映射。因为为了最终能潜在的控制负载分发，现用第二映射（IP地址到主机的映射），对所有的尝试（是否提交或执行）主要用IP地址向主机的映射。在引入的分布式请求用到了RR-DNS（轮叫域名服务），考虑了两个负载均衡的运算法则，无状态的和有状态的。无状态的方式简单的用

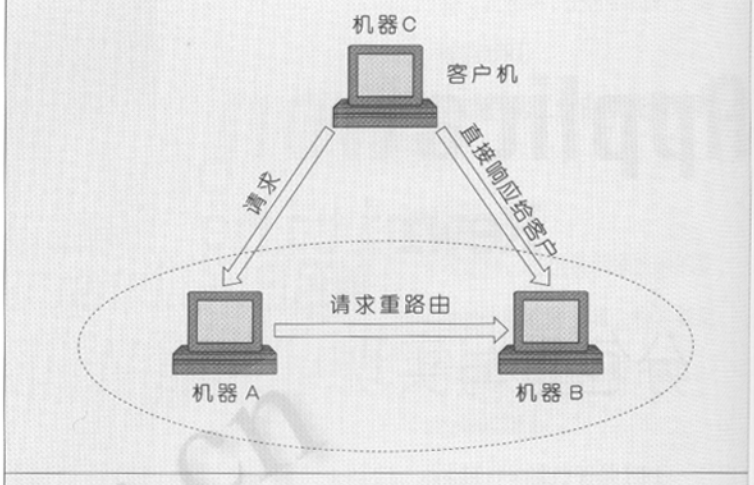

图1 执行过程

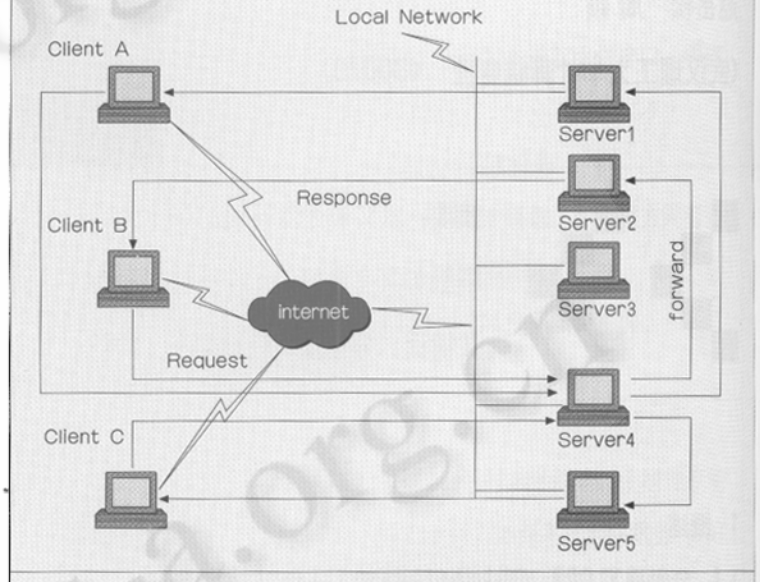

图2 集中连接路由

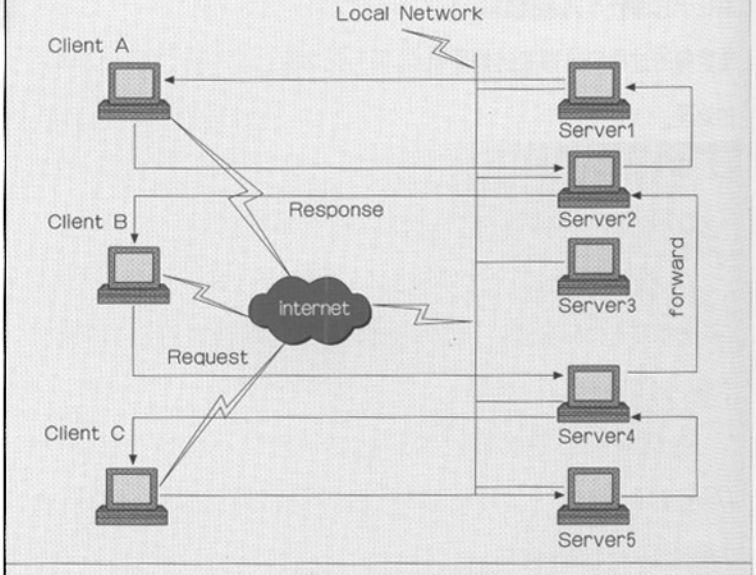

图3 分布式连接路由

了原IP地址和端口地址去计算哈希表，并选择一个服务器；有状态的方式利用当前用IP伪装发送的负载信息去选择目标服务和连接。实验表明RR-DNS无状态方式执行过程像一个完全依据系统的吞吐量和延迟性能来平衡负载的负载平衡器。

2.3 分布式连接路由

DPR技术有时也叫分布式连接路由。不象已有的解决方法，(如图2)依赖于单一集中的连接路由，DPR允许在IP地址与主机分布式执行过程中映射，成为现在有效的可伸缩性技术的时尚，尤其是DPR可以看作是一个在IP地址向n个服务器映射的分布式方法，用DPR集群Web服务中每一个主机既可以作为一个服务器也可以用为一个连接路由。

DPR可以同时进行服务和对集群服务中所有主机共享路由响应(如图3)。分布式连接路由功能上达到了真正的可伸缩性，因而对集群增加一个新的主机自动地增加足够的能力以促进Web服务和连接路由的能力。

另外，在web站点紧急任务时，在连接路由和连接服务之间集中连接路由的能力趋于明显的不平衡。而这些问题在基于DPR的集群构架中得以避免，增加的机器消除了单点失败情况，也就在增加系统能力的同时既提供了可伸缩能力连接路由的能力也增加了连接服务的能力。

现有两种集群：单层和两层。它们除了在网络连接构架中的区别外，就是在单层构架（如DPR）里所有的节点可等同如图1中A节点，能够同时作为负载均衡器(前端节点)，或者作为文件服务器（后端节点）进行操作；在两层构架中每一节点明确的配置了作为前端节点还是后端节点服务。在单层构架中如分包重写(DPR)，没有一个集中的负载均衡器而是用一个分布式负载平衡方法。一个服务传送一个HTTP重定向消息给客户，用一个IP伪装设计在集群服务中去执行分包重写。

2.4 路由表

考虑性能因素，机器的重定向连接的机制在内核中执行。当决定重定向或者服务一个新的连接且当要连接的进入包被重定向时，对速度的要求显得尤为重要。

当一个机器接收到一个IP包，内核调用函数ip_receive(),做一定的修改可进行重定向连接。这里，IP包是被检测过的，如果它包含TCP包，且这个TCP的目的端口是80端口（或者web服务器正在运行），我们知道它是一个http连接，并且是直接从客户端进来的。如果TCP包包含一个SYN，那么我们可知一个新的连接被请求，现要决定服务或者提交它。如果这机器在门槛值以下，或者当前的负载在整个系统中是最低的，那么这个请求被保存在本地且路由表不被升级。如果当前的负载超过了门槛值，并且相对其它机器负载是最低的，那么路由表将升级并将该包提交。如果包不是SYN，那么我们查找路由表中如果连接被重定向，则包被提交。如果IP包包含一个IP-IP包，我们知道它是一个用于重定向的包，且我们必须为它服务，我们解开IP-IP包并传送TCP包给TCP层处理。而利用未用的碎片弥补位，我们能检查到源IP地址是否符合服务器在DPR检查重定向连接中的分担情况。(这整个过程中有三个函数加到内核中，它们是dprsocketcall.h，dpr.h，dpr.c，用来执行查询表和负载表)。

路由表的完成也就是整个系统中的请求→连接→重定向整个过程的实现。

3 总结

判定一个集群连接的网络有几种因素，如节点中的耦合程度，信息安排，所有的节点在主客关系上是否相称等。而DPR这个分布式处理请求事务的方法，允许所有主机参与请求过程的重定向，具有分布式连接路由方式，系统中大量路由能力使得其具有很强的可伸缩性，另外DPR能合理的安排信息的传输和处理情况使得所有节点具有良好的耦合关系，形成客户与服务的协调与匀称的合理机制，还为系统提供了更好的系统性能价格比。运用DPR的集群服务可获得良好可伸缩性的集群服务系统。

参考文献

1 Luis Aversa，Azer Bestavros:Load Balancing a Cluster of Web Servers Using Distributed Packet Rewriting http://cs-www.bu.edu/~best/res/projects/DPRClusterLoadBalancing/。

2 CSE 820 Term Project Spring 2001- Survey of Clustered Web Server Architectures Amit Sahoo-Saravanan Palanisamy www.cse.msu.edu/~sahooami/cse820paper.doc。

3 Azer Bestavros, Mark Crovella ,Jun Liu&David Martin :Distributed Packet Rewritingand its Application to Scalable Server Architectures。

4 Luis Aversa and Azer Bestavros. Load Balancing a Cluster of Web Servers Using Distributed Packet Rewriting. In Proceedings of IPCCC'2000: The IEEE International Performance, Computing, and Communication Conference, Phoenix, AZ, February 2000。