

内容分发网络请求路由研究

惠 雯^{1,2} 尹 浩² 林 闯² 杨 扬¹

(北京科技大学计算机与通信工程学院 北京 100083)¹ (清华大学计算机科学与技术系 北京 100084)²

摘 要 内容分发网络的出现解决了传统“尽力而为”的 IP 网络与当前多种新应用需求之间的矛盾。请求路由是内容分发网络的一个核心技术,它通过有效定位服务资源,可使用户快速有效地得到内容或服务,实现全局负载均衡。一直以来,请求路由都是学术界和工业界关注的热点,但目前还没有系统地归纳和总结该领域的发展现状。为致力于这项工作,深入研究了请求路由的相关算法和机制,并结合当前内容分发网络面临的复杂环境及需要服务的各种应用,对该领域有待进一步研究的方向和可能的解决方案给出一些建议。

关键词 内容分发网络,请求路由,资源定位,请求重定向

中图分类号 TP393 **文献标识码** A

Request Routing in Content Delivery Networks

HUI Wen^{1,2} YIN Hao² LIN Chuang² YANG Yang¹

(School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China)¹

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)²

Abstract The technique of Content Delivery Networks(CDNs) has emerged and been developed for bridging the gap between the emerging applications and best effort IP networks. Request routing is a key technology of CDN, and is responsible for routing user requests to an appropriate server, so that the services can be delivered to users effectively and efficiently, realizing the global server load balancing. Request routing has been a hot issue of academic and industrial circles in recent years. However, there are fewer works on analyzing and summarizing the related research about this topic. Thus, this paper was working on it. We systematically surveyed the existing request routing algorithms and mechanisms. Furthermore, in terms of the CDN environment and the varieties of applications, we addressed the challenges faced, and presented the research tendencies and the potential solutions.

Keywords Content delivery network, Request routing, Resource location, Request redirection

1 引言

随着计算机和通信技术的快速发展,以及各种新应用的出现,网络环境日益复杂,对网络资源的需求迅速增加。CNNIC 报告显示,截至 2010 年 6 月底,我国网民规模已达 4.2 亿,较 2009 年底增加了 9.4%。如何缓解网络拥塞、提高网络传输的速度和质量,已成为困扰众多服务提供商的一大难题。目前 Internet 的基础架构是基于包交换的 IP 网络,采用端到端的设计原则,提供简单的“尽力而为”服务。这一架构虽然易于实现,具有较强的可扩展性,但却难以满足许多新应用的需求,特别是一些多媒体应用的需求。内容分发网络(Content Delivery Network, CDN)的出现有效地解决了这一问题^[2,35]。CDN 通过在传统的 IP 网络上部署服务节点,并利用应用层协议将这些服务节点组织成一个构建在 IP 网络之上的覆盖网络,为用户提供高效灵活的内容分发服务。终

端用户可从最合适的服务节点获取内容,这样既提高了服务质量,又缓解了骨干网的拥塞。同时,这一架构是在应用层解决问题,无需改变底层网络结构与骨干设备的配置,因而具有较强的灵活性和可扩展性。

请求路由(Request routing)是 CDN 中的一个核心技术。与传统路由不同,请求路由的目的是为用户的内容请求定位最佳服务节点,以使用户快速有效地得到内容或服务,实现全局负载均衡。具体来讲,请求路由的过程可分为两个阶段:一是资源定位,即根据用户请求的内容和网络条件,为其确定最佳服务节点;二是请求重定向,即根据资源定位的结果,将用户请求引导到最佳服务节点,由该节点为用户提供服务。最佳服务节点的选择由诸多因素决定,如用户位置和可用网络资源等。图 1 是一个高度概括的 CDN 请求路由模型。对终端用户来讲,请求路由的整个过程都是透明的^[3,4],它既可以部署在客户端和/或服务端,也可以部署在网络中。而且,

到稿日期:2011-03-06 返修日期:2011-05-11 本文受国家自然科学基金(60873254, 61003226), 973 国家重点基础研究发展计划(2011CB302600), 教育部新世纪优秀人才支持计划项目资助。

惠 雯(1982—),女,博士生,主要研究方向为多媒体通信, E-mail: hwen97@126.com; 尹 浩(1974—),男,博士,副教授,博士生导师,主要研究方向为计算机网络、多媒体通信与安全、网络计算等; 林 闯(1948—),男,博士,教授,博士生导师,主要研究方向为计算机网络与系统性能评价; 杨 扬(1955—),男,博士,教授,博士生导师,主要研究方向为多媒体通信、网络计算、无线通信、图像处理与模式识别等。

请求路由的实现也不再局限于网络层,很多请求路由的算法和机制都是基于应用层来实现的。

CDN 请求路由一直以来都是学术界和工业界关注的热点,已取得很多有意义的研究成果,但目前还没有系统地归纳和总结其发展现状。本文将致力于这项工作,首先对现有研究成果进行分析和总结,然后在此基础上,结合当前 CDN 面临的复杂环境及需要服务的各种应用,对该领域有待进一步研究的方向和可能的解决方案给出一些建议。

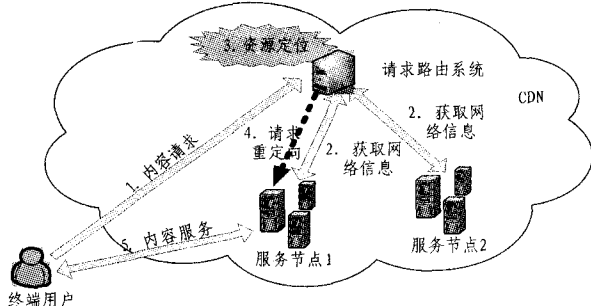


图1 CDN 请求路由模型

2 CDN 请求路由的分类

CDN 请求路由有多种分类标准。按照应用的地理结构的不同,可分为全局负载均衡和本地负载均衡^[37]。全局负载均衡是对放置在不同地理位置、有不同网络结构的服务器群做负载均衡,使用户以最快的速度访问到离自己最近的服务器群;而本地负载均衡则是对本地服务器群做负载均衡,根据网络状况和服务器负载,将用户请求在不同服务器间进行调配。这种分类方法较适合于传统 CDN 中央控制、服务器分层且分区思想。

如果以 OSI 参考模型为标准,可分为基于 DNS 请求路由、传输层请求路由和应用层请求路由^[8,9]。基于 DNS 请求路由是 Internet 上广泛使用的路由技术,同样它也适用于 CDN;传输层请求路由可与基于 DNS 请求路由一起工作,提供细粒度的请求路由;而应用层请求路由则可看作在传输层请求路由基础上更细粒度的请求路由。此外,文献^[8]中还提出了内容层请求路由,它也称为基于名字路由,根据用户请求的内容的名字(并不是完全的 URL)进行路由。按照 OSI 参考模型进行划分的方法继承了传统路由划分方法的思想,但由于 OSI 模型本身的局限性,该方法还不能完全体现 CDN 请求路由的特点。

还有一种分类方法以请求路由的过程为依据,即按照资源定位和请求重定向两个阶段来划分各种请求路由算法和机制^[3,38]。这种分类方法能够比较清晰地描述 CDN 请求路由的原理,还能更好地体现请求路由的发展趋势,因此,本文将基于该方法对相关算法和机制进行分析和综述。

3 资源定位

资源定位是一个比较复杂的问题,因为网络和服务器的状态是实时变化的,很难找到一种通用的准则来适应所有情况。也就是说,对于同一用户的内容请求,在不同时刻,适合为其提供服务的节点未必是同一个节点。即使有两个用户在同一时刻请求内容,适合为他们提供服务的节点也不一定就是同一个节点。Rodriguez 等^[17]曾提出一种方法,即通过同

时从多个服务节点获取内容来绕开资源定位的问题。但 Kangasharju 等^[16]却发现,在采用这种方法时,各个服务节点间通信所造成的延迟已超过回避资源定位所节省的时间。因此,从目前研究来看,资源定位仍是一个亟待解决的重要问题。

资源定位算法有不同分类标准。根据 CDN 控制模式的不同,可分为中央控制算法、分布式控制算法,以及基于中央控制和分布式控制的混合算法^[1];根据度量准则的不同,可分为基于距离算法和基于负载算法^[26];根据算法是否具有自适应性,又可分为自适应算法和非自适应算法^[3,6]。自适应算法根据当前网络状况和服务器工作状况进行定位,准确性更高,稳定性更好,但往往比较复杂;非自适应算法大多采用启发式算法,实现起来比较容易,特别是能够满足假设条件时,其效率能达到很高,但通常需要保存大量历史数据。

3.1 非自适应资源定位算法

表 1 总结了一些典型的非自适应资源定位算法。其中轮询法不需要任何度量准则,算法最为简单,但由于没有考虑任何服务器信息和网络信息,因此定位结果也最盲目。其他几种算法都根据服务器状况或网络状况,采用一定的准则来定位,因此对于所用准则而言具有最佳性。表 1 中所列算法还存在一系列变型。例如,Delgadillo 等^[14]在轮询法的基础上提出了权重轮询法,即根据每台服务器的处理能力为其设置不同的权重,使分配给每台服务器的用户连接数与其权重成正比,解决了服务器处理能力的异构问题;Rabinovich 等^[13]不仅根据已处理的请求数来判断各节点负载,还通过实时监测各节点的负载信息,将二者结合作为选择依据;Wang 等^[6]基于一致性哈希算法和最高随机权重算法,将用户请求映射到多个节点上,提出了 R-CHash 和 R-HRW 算法,并进一步结合节点负载信息,提出了 LR-CHash 和 LR-HRW 算法。

3.2 自适应资源定位算法

表 2 总结了一些典型的自适应资源定位算法。自适应算法通常会考虑服务器状况或网络环境的变化,因此,相对非自适应算法而言,其具有更好的可扩展性和鲁棒性,以及更高的效率,但该算法的实现也更加复杂。另外,在应用自适应算法时,需要一定的方法来测量网络状况和服务节点的实际工作状态。常用方法有 3 种:主动探测法、被动测量法和服务器反馈法^[9,38]。其中前两种方法可用于测量路径距离、时延和带宽等,主动探测法针对性较强,效率较高,但这会增加网络流量,甚至引发安全问题;被动测量法不会增加额外流量,在实时性和准确性方面也更具优势,但不够灵活。第三种方法专门用于获取服务器负载信息,可通过静态测量、动态测量或静态与动态相混合的方式来实现。

在表 2 的各种自适应算法中,基于往返时间的算法是最直观和常用的,它能够较好地反映客户端实际感受到的服务质量,但这会消耗大量资源度量各服务器间的时延,因此出现了一些改进算法,如利用网络坐标^[27]将各服务器映射到 n 维空间中,在坐标空间中进行距离测算。此外,采用单一度量准则的定位结果也往往不够理想,因此又出现了一些结合多种度量准则的自适应算法。如 Cisco 公司的 DistributedDirector^[14],基于自治域间和自治域内距离以及时延这 3 个参数的加权和来度量网络距离,相比单一参数,具有更高的精度和更大的灵活性,但需要部署大量附加设备来测量数据,导致复杂

度增加,开销增大,而且其采用主动探测法测量数据,会引入额外流量。Akamai的映射算法^[28]综合考虑了网络距离、端到端时延、丢包率、服务器负载等多种因素,具有极强的自适应能力和全局负载均衡能力,且效率极高;但由于参数过多,

计算过程非常复杂,并且需要部署很多网络探测设备和全局信息管理设备等附加硬件设备,因此导致成本过高。此外,针对多媒体内容分发,一些算法还强调了与客户端 QoS 需求及其他应用需求^[4]的结合。

表1 非自适应资源定位算法

资源定位算法	度量准则	计算方法	优点	缺点
轮询法 ^[10]	无	按照预先定义好的次序将用户请求轮流定位到不同的服务节点(假设所有节点的处理能力相同或相近,且它们中的任意一个均可响应用户请求)	算法简单,在较小范围内对服务器集群进行资源定位时效率较高	未考虑各服务节点状况和网络状况,很难保证所选节点是最佳服务节点,难以实现负载均衡
基于距离算法	物理距离	客户端与服务节点在实际中的物理距离	当服务节点的负载不是很高,且网络状况不是很差时,效果较好	很难准确反映网络时延
基于负载算法	网络负载 ^[36]	根据网络拓扑构建负载平衡树,对于不同的内容请求,对树进行相应调整,使其整体负载达到平衡	可实现全局负载均衡	未考虑网络状况,适应能力较差,且会增加路由器负担
	服务器负载 ^[7]	根据各服务节点已处理的用户请求数来判断各节点负载,由此选择最佳服务节点(假设节点负载只与用户请求数有关)	可使用户请求比较平均地分配到各服务节点,在一定程度上缩短了网络时延,具有一定的适应能力	未考虑网络状况,难以保证所选节点是最佳服务节点。且未考虑服务器处理能力的异构问题
模 N 哈希算法	服务节点数	按照传统哈希算法,将用户请求内容的 URL 对节点的个数取模,由所得哈希值所对应的节点向用户提供内容	效率较高	当节点数量发生变化时,需重新定位所有用户请求
一致性哈希算法 (Consistent Hashing, CHash) ^[15]	服务节点数	采用某种哈希函数,将服务节点的 IP 地址或名称映射为一个首尾相接的圆环,然后将用户请求内容的 URL 同样映射到该环中,由顺时针方向哈希值最接近 URL 的 ID 所对应的节点为用户提供内容。若该节点中没有所需内容,则由环上的下一个节点为用户服务,以此类推	效率较高,开销较少;当节点数量发生变化时,只需重新定位部分用户请求	比模 N 哈希算法复杂
最高随机权重算法 (Highest Random Weight, HRW) ^[49]	服务节点数	将用户请求内容的 URL 和服务节点的名称一起作为一个伪随机数发生器的种子,用来生成一个权重。对于每一个用户请求内容的 URL,由权重最大值所对应的节点为用户提供内容	当节点数量发生变化时,对资源定位的影响很小,可扩展性较好;有助于减少转向器间的内部通信	比一致性哈希算法更复杂,效率略低,实现代价更高

表2 自适应资源定位算法

资源定位算法	度量准则	计算方法	度量方法	优点	缺点
基于网络距离算法	路由跳数 ^[32]	从客户端到服务器端经过路由器的个数		对路由的改变具有较好的适应性	需配置路由器,导致开销增大,且定位不够准确
	自治域距离 ^[10,33]	将每个自治域看作图的顶点,图的边代表自治域间交换信息的路径,客户端与服务节点之间的距离等于它们所在自治域间最短路径所对应的边数	主动探测法,被动测量法	算法简单,易于实现	不同自治域大小相差很大,很难保证定位的准确性
基于丢包率算法 ^[5]	丢包率	从客户端到服务器端的丢包率	主动探测法,被动测量法	效率较高,易于实现	准确度不高
基于时延算法	TCP 连接时间 ^[25,20,21]	在服务器端测量建立 TCP 连接所用时间(假设 TCP 连接时间与服务质量直接相关,反映了服务器处理能力、服务器负载和网络负载等诸多要素)	被动测量法(通常对客户端进行聚类,用聚类的特性来代表其包含的所有终端用户,可节约带宽和响应时间,有利于资源利用)		
	DNS 查询时间	本地 DNS 服务器从发出查询请求到收到服务器端 IP 地址所用时间		效率较高,复杂度小	容易受各种因素影响,且有时需要维护一个中心数据库,限制了系统的可扩展性
	往返时间(Round-Trip Time, RTT) ^[29]	客户端发出请求到接收到服务器端的第一个字节所用时间	主动探测法,被动测量法		
	内容传输时间	客户端完整接收内容所用时间			
	响应时间(Response Time, RT) ^[31]	客户端发出请求到收到所有内容所用时间			
基于负载算法	网络负载 ^[12]	可用带宽(同时还要考虑最大带宽)	主动探测法,被动测量法	准确性较好	未考虑服务器负载
	服务器负载 ^[31]	(1)基于各种性能参数,如 CPU、硬盘空间、内存、连接数、I/O 等; (2)基于服务器总体性能,如吞吐量、响应时间等	服务器反馈法	具有较好的实时性和准确性	未考虑网络状况

4 请求重定向

根据请求处理过程的不同,可以将请求重定向分为:负载均衡重定向、DNS重定向、HTTP重定向、URL重写重定向、Anycast重定向和对等CDN重定向。

4.1 负载均衡重定向

在负载均衡重定向方法中,每个服务节点都配有一个负载均衡交换机。该交换机有两个用途,一是获取本地各服务器的健康性和负载状况等,二是与其他服务节点的负载均衡交换机交换信息,了解其他节点的位置和性能信息。同时,负载均衡交换机还可用作某个特定区域的智能授权DNS。以Nortel Networks为例^[39],当用户请求被发送到某个智能授权DNS(即负载均衡交换机)时,该DNS可根据本地各服务器及其他各节点的情况,利用某种资源定位算法,返回最佳服务节点的IP地址,并最终将用户请求定位到最佳服务节点中负载最轻的服务器。此外,Foundry Networks^[40]还考虑了负载均衡交换机与终端用户的距离。

这种方法最大的好处是,负载均衡交换机不仅能够获取本地服务器的状况,还能收集其他服务节点的情况,也就是能够在全局范围内选择最佳服务器,并且还不用在网络中增加额外设备。其缺点是,需要人工配置各服务节点的负载均衡交换机。

4.2 DNS重定向

DNS重定向通过修改授权DNS来实现各服务器到IP地址的映射。授权DNS由CDN服务提供商控制,当用户请求内容时,授权DNS会返回所有包含所请求内容的服务器的IP地址,并根据当前网络状况、负载和临近性等信息从中选择一台最佳服务器向用户提供内容。如果授权DNS返回多个服务器的IP地址,客户端本地DNS可循环使用这些地址,将请求相同内容的不同用户定位到不同的服务器^[8,9],这种方法增强了CDN的可靠性和负载均衡。

DNS重定向在实际中应用广泛,不仅因为其原理简单,可借鉴已有DNS路由技术,利用现有设备,还因为它具有一定的独立性,不会影响Internet上其他应用的DNS解析过程^[41]。著名的CDN提供商Akamai采用的就是这种重定向方法^[47]。不过,DNS查询次数过多会造成网络延迟增加。常用的解决方案是多层DNS解析法^[8,9],该方法利用NS和CNAME记录,通过在单服务器解析过程中合理利用多个请求路由DNS,能够做到从多个具有不同规则的路由DNS中选择一个来响应用户请求。

DNS重定向还存在以下问题:计算客户端与服务器之间的距离时,近似用客户端本地DNS的位置代替客户端的位置,有可能导致定位结果不准确;不考虑客户端IP地址还导致DNS查询的可扩展性较差,而且限制了对客户端的认证控制;通常为返回给用户的记录设置较短的生存周期,以便对网络变化快速做出响应,但这会造成用户频繁发送请求,增加DNS的负载;使用网络级度量不会响应应用级的失效请求,客户请求可能被连续重定向到一台无反应的服务器。

4.3 HTTP重定向

HTTP重定向是用HTTP头来传输服务器的信息,需要客户端参与重定向的处理。HTTP协议允许Web服务器以一个特定的消息来响应用户请求,将用户请求重定向到其他

服务器。利用这一思想,可用一个专门的服务器来接收用户的内容请求,然后根据某种资源定位算法,以HTTP重定向的方式将用户请求定位到最佳服务器上。具体地讲,HTTP重定向分为基于URL、基于Mime头和基于特定标识符3种路由方式^[8,9]。基于URL的路由方式利用所请求内容的URL或URL前缀来实现重定向,包括302-重定向和In-path要素两种方法;基于Mime头的路由方式在进行重定向时使用Mime头信息,如Cookie、语言、用户代理等;基于特定标识符的路由方式则是利用一个特定标识符(如SSL会话标识符)与用户建立连接,完成请求重定向。

HTTP重定向的主要优点是简单和灵活,很容易被应用,而且还能实现比较精细的内容管理。但其可扩展性较差,使用该方法时,由于终端用户在第一次通信时必须与同一台服务器进行通信,这样随着终端用户请求数量的增加,会造成瓶颈现象。该方法最大的弊端是缺乏透明度,而且需要传送额外消息,容易影响效率。

4.4 URL重写重定向

URL重写重定向是CDN在应用层的最早使用,也是当时比较流行的一种请求重定向方法,但现在已经很少使用。其核心思想是终端用户请求被直接定位到内容源服务器,由源服务器进行内容分配。当源服务器收到终端用户的请求后,按照某种资源定位算法来确定最佳服务器,然后对URL进行修改,使其指向最佳服务器,重写后的URL将嵌入到请求响应中,返回给终端用户,终端用户通过该URL向给定的服务器请求服务。

该方法既可用于一般性的内容分发,也可实现特定内容的分发,如点播分发等。对应地,有两种路由方式,即主动URL重写和按需URL重写^[3,8,9]。主动URL重写是在请求被定位到源服务器之前重写嵌入的URL,也就是预先存储一些重定向方案,这样不同的客户端可以被定位到不同位置的服务器^[38]。这种方式一般不允许考虑用户的特殊需求。而按需URL重写则是当请求到达源服务器时,修改相应的URL,由于此时终端用户的身份已通过认证,因此在定位最佳服务器时可考虑不同用户的具体要求。这种方式也称为动态URL重写。

通过URL重写,内容提供商在不改变网络结构的情况下,可不通过其他转换设备,而直接与终端用户建立联系,将请求引导到最佳服务器上。该方法最大的好处是,能够实现所请求对象的动态组合,提高内容分发效率;而且与DNS重定向相结合,还可实现精细重定向。当然该方法也存在缺陷,比如客户端的第一个请求都必须由源服务器来响应,这就限制了该方法的应用范围;另外URL解析也会导致时延增加。值得注意的是,在对服务器的URL重写进行缓存时容易出现,它可能会指向一个已经不可用或不合适的服务器,解决方法是不对其进行缓存,或设置较短的缓存时间。

4.5 Anycast重定向

Anycast最初的定义是,用一个Anycast地址标识一组提供某种服务的主机,发送到该Anycast地址的数据报将被投递给这组主机中的任意一台,这是一种无状态的、尽力而为的投递服务。直到IPv6协议正式接纳Anycast服务,Anycast才被赋予了新的含义,即允许将数据报投递给共享同一Anycast地址的接收者中最近的一个^[42,43]。这也是CDN中

基于 IP 层 Anycast 重定向的基本思想。

Anycast 重定向可分为两类: IP 层 Anycast 重定向和应用层 Anycast 重定向。IP 层 Anycast 重定向假设同一个 IP 地址可以被分配给若干节点,每个 IP 路由器在其路由表中维护着一条最接近该路由器的节点的路径,因此经过不同 IP 路由器的用户请求会被定位到这些具有相同 IP 地址的不同节点。IP 层 Anycast 重定向可在异构平台上很好地实现内容服务,但如果用单播路由协议路由 Anycast,由于 Anycast 并不遵守单播路由协议的层次汇聚规则,共享一个 Anycast 地址的节点可能分散在不同地方,因此每个全局 Anycast 地址必须作为独立的路由表项处理,这就使得路由表随着 Anycast 组数呈比例增长,造成可扩展性问题。Katabi 等^[45]提出了一个 Anycast 的可扩展结构 GIA。在该结构中,区域边界路由器通过分析区域间 Anycast 路由特性,把域间 Anycast 路由分成两部分来满足可扩展性要求。

IP 层 Anycast 重定向只能使用 IP 层的路由跳数来定位最佳节点,而应用层 Anycast 重定向则能根据更具体的性能指标来定位最佳节点,如服务器处理能力、服务器负载、网络距离、网络负载等。Fei 等^[44]通过一系列执行 Anycast 域名到 IP 地址之间映射的 Anycast 解析器来实现应用层 Anycast 重定向,客户端产生的 Anycast 查询传送给 Anycast 解析器处理,Anycast 解析器利用推和探测技术,收集和维持 Anycast 组中每个节点的性能信息,然后采用一定的资源定位算法选择合适的节点反馈给客户端。Gritter 和 Cheriton^[46]提出的应用层 Anycast 重定向方法又称为内容路由方法,由扩展到支持命名的内容路由器来执行,他们设计了因特网名字解析协议(INRP),可实现快速名字查找,还设计了基于名字的路由协议(NBRP)来执行路由,而且 NBRP 还支持路由聚合。应用层 Anycast 重定向比 IP 层 Anycast 重定向具有更大的灵活性,但需要收集更多信息,如服务器和链路的负载信息等,从而导致成本增加。

4.6 对等 CDN 重定向

为了更加快速有效地将内容分发给终端用户,打破传统 CDN 分级控制模式的局限,人们开始关注于 CDN 的互联^[48]。通过 CDN 提供商之间的合作,使每个 CDN 可以借助其他 CDN 的服务器为用户提供质量更高的服务,使其具备更好的容错能力,同时还能减少设备成本^[30]。目前对等 CDN 重定向方法主要有 4 种:集中式目录法、洪泛法、分布式哈希表(Distributed Hash Table,DHT)法和混合式路由法。

在集中式目录法中,存在一个中央节点用来发布所有服务节点的资源信息,但所有服务节点均与中央节点通信,从中央节点获取所需资源信息^[51]。当多个服务节点包含所需内容时,中央节点将根据某种资源定位算法,从中选择一个最佳服务节点反馈给请求节点^[3]。这种方法的优点是原理简单,易于实现,且能够很好地进行资源分配。但其受中央节点处理能力的限制,可扩展性较差,且中央节点不能出错。

洪泛法不同于集中式目录法,它不需要预先知道各服务节点的资源信息,每个来自服务节点的内容请求都会被发送到与该节点直接相连的其他节点,如果这些节点均没有所需内容,则请求会继续传播给与这些节点相连的节点,以此类推,直到找到所需内容或遍历所有节点^[52]。该方法在网络较小时效率很高,但会造成网络流量急剧增加,占用大量带宽,

而且随着网络规模的扩大,可扩展性也存在问题。

DHT 法是目前可扩展性最好的方法,采用纯分布式的消息传递机制,根据关键字进行资源定位。它为每个服务节点分配一个唯一的节点 ID,资源对象通过 hash 运算产生一个唯一的资源 ID,每个资源都存储在节点 ID 与其资源 ID 相同或相似的节点上,这样当需要查找某个资源时,采用同样算法就可定位到存储该资源的节点上^[50]。该方法对较大网络范围内的请求路由很有效,且更有利于各节点间的负载均衡,但实现起来比其他方法要复杂。为精确定位所需内容,可采用表 1 中的一系列哈希算法。

混合式路由法结合了集中式目录法和洪泛法的优点,选择性能较高的服务节点作为授权节点,在各授权节点上存储部分其他节点的信息,资源定位算法仅在授权节点间转发,由授权节点将内容请求发送给最佳服务节点^[34]。这种方法继承了洪泛法的分布式结构,可扩展性较好,虽没有中间节点索引资源信息,但各授权节点构成的覆盖层提供了很好的索引功能,使搜索效率大大提高。其缺点是对授权节点的依赖性较大,容易被攻击,容错性也会受到影响。

5 分析与讨论

5.1 研究进展

从 1998 年至今,CDN 经历了以 Akamai 为代表的分布式 CDN、以 Limelight 为代表的高速网辅助 CDN、以 CacheLogic 为代表的对等网(Peer-to-Peer,P2P)辅助 CDN 和云计算模式 CDN 这 4 个发展阶段,其始终致力于为互联网内容分发提供高效便捷的解决方案。在 CDN 发展的不同时期,根据内容分发任务的不同特点,所采用的请求路由技术也有所区别。

(1)第一代请求路由。CDN 发展之初,网络规模不大,分发的内容主要是静态文本和图片等,数据量都比较小。CDN 服务提供商通过高度分布式的服务器部署方式来构建 CDN。这一时期的请求路由技术主要解决内容请求的快速响应问题。资源定位算法基本上是沿用一些传统路由算法的思想,如轮询法和基于距离算法等。随着对内容需求的增加,网络规模不断扩大,服务器的分布和用户请求的随机性使得已有算法造成网络负载不均,服务器响应过慢。于是有学者提出一些改进算法和新算法来解决上述问题,如基于网络负载的算法可实现全局负载均衡,基于服务器负载的算法能大大提高系统响应速度。请求重定向方法起初在已有重定向技术的基础上进行改进,主要有 DNS 重定向和 URL 重写重定向两种。随着 CDN 的发展,相继又出现了 HTTP 重定向、Anycast 重定向和负载均衡重定向。

(2)第二代请求路由。20 世纪初,各种多媒体内容(如音频、视频等)开始占用 Internet 的主要带宽,CDN 开始关注于大数据量多媒体内容的高效分发,通过在 Internet 中部署少量大服务节点,然后利用私有高速网络将这些大节点相连接,以保证充分的带宽和时延。这一时期的请求路由技术主要解决如何在网络环境多变的情况下顺利完成内容分发任务,即鲁棒性,以及如何实现网络扩展环境下的内容分发,即可扩展性等方面的问题。资源定位算法在第一代请求路由的基础上得到进一步改进,如 Cisco Distributed Director 的自适应算法^[14]、Akamai 的自适应算法^[28]等。这一阶段的请求重定向方法主要以提高引导效率和保证整个 CDN 负载均衡为核心

思想,表现为改进的 DNS 重定向,还有负载均衡重定向。前者主要解决 CDN 快速扩展所造成的智能 DNS 负载过重问题,后者则是通过一定的测量手段,使网络中的一些设备能够获取整个网络信息,以便请求到达时可根据网络全局信息将请求引导到最佳服务器,从而实现全局负载均衡。请求路由发展到第二代时,现有的资源定位和请求重定向的各种算法和机制基本上都已出现。

(3)第三代请求路由。近年来,用户对以视频为主的流媒体内容的需求不断增加,且要求也不断提高,对这些内容的分发会消耗大量网络资源,特别是带宽,严重影响了 CDN 的服务质量。为此,学术界和工业界开始关注并致力于结合 P2P 和 CDN 的优势,构建 P2P 辅助 CDN 体系结构。根据 CDN 与 P2P 在网络中结合的位置不同,可将 P2P 辅助 CDN 分为终端用户型 CDN-P2P 和节点型 CDN-P2P。

对于终端用户型 CDN-P2P,请求路由结合了 DNS 重定向和对等 CDN 重定向。收到用户请求后,当采用传统 CDN 模式时,系统用 DNS 重定向方法直接将请求定位并将其引导到最佳节点;当转到 CDN-P2P 模式时,系统先将请求定位并将其引导到最佳节点,然后该节点将终端用户加入到已组成的 P2P 网络,并服务于该网络中所有的终端用户^[22]。

对于节点型 CDN-P2P, Jernberg 等^[19]提出了一种叫做 DOH 的 CDN-P2P 网络,该网络的 DOH 节点相互连接成一个结构化 P2P 网络,请求路由结合了负载均衡重定向和对等 CDN 重定向中的 DHT 法。DNS 根据当前负载和网络条件将请求引导到 DOH 节点,节点若能够服务,就直接服务;否则,节点利用 DHT 法定位并获取资源,然后服务于终端用户。Fujita 等^[53]提出了两种算法来实现 CDN 和 P2P 的结合:静态算法和动态算法。其中,静态算法采用 Anycast 和 DHT 相结合的方法,即在 CDN 上层使用 Anycast 来确定最佳节点,被选择的节点如果包含所需内容,就直接服务,否则采用 DHT 法来定位资源。动态算法先使用一些测量技术获得一定的网络信息,然后采用静态算法,并根据一定的准则选择更适合的资源,从而达到优化系统的目的。

(4)第四代请求路由。云计算(Cloud computing)模式 CDN 是 CDN 的一个新的发展方向。云计算开启了 CDN 网络架构设计和应用的新思路,一个典型应用就是在解决流媒体分发问题时,利用云计算模式 CDN 可以在保证分发效率的同时,提供可靠的 QoS 保障。这一阶段的请求路由需要对网络环境和计算环境具有很强的适应性。特别是随着手机等移动终端用户逐渐成为网络内容消费的主体,一些相对静态的重定向方法也显得力不从心,找到更加有效的请求路由方法是 CDN 服务满足未来新应用需求的一个必要条件。

5.2 研究挑战与未来展望

请求路由发展到今天,已在前两代 CDN 架构中取得了巨大成功。当第一代 CDN 为克服传统 C/S 模式的局限性而出现在用户面前时,请求路由解决了如何在资源分布不均的网络中定位最佳服务器的问题。当 CDN 进入第二代,需要将众多小型 CDN 组合成更大的 CDN 时,网络环境也不再像小型 CDN 那样单一,分发的内容变得更加复杂,这时的请求路由关注的是鲁棒性和可扩展性问题,于是出现了诸多自适应资源定位算法。第三代 P2P 辅助 CDN 目前尚处于发展阶段,其目标是实现高清、流媒体等极耗资源和带宽的内容分发,请

求路由研究的重点是如何将 CDN 与 P2P 中不同的路由技术很好地融合在一起。如今,又出现了第四代云计算模式 CDN,旨在通过网络把多个成本相对较低的计算实体融合成一个具有强大计算能力的完美系统,并借助 SaaS、PaaS、IaaS、MSP 等先进的商业模式把这一强大的计算能力分发到终端用户手中^[24]。此时的请求路由在保证效率的同时,更注重提供可靠的 QoS 保障。

以下总结了目前请求路由技术所面临的几个关键性问题及未来可能的发展方向。

(1)适应复杂多变的网络环境。随着各种新型网络终端和接入方式的出现,CDN 面临的网络环境日益复杂。同时,各种新应用的发展也对请求路由的实现提出了新的挑战。在复杂多变的网络环境下,受测量技术限制,有时很难准确获取各服务节点或网络的状态信息,从而导致资源定位结果发生偏差,影响整体服务质量。最近有研究^[23]提出结合网络坐标的解决方案,在以往利用网络坐标测量距离的基础上进行改进,实现对网络负载和服务器负载的测量,为准确定位最佳服务器提供了新的研究思路。

(2)克服现有方法的局限性。目前大多数请求路由方法仍是基于 DNS 架构的,这种方式对网络环境和计算环境的适应性较差。随着手机等移动终端用户逐渐成为网络内容消费的主体,这种相对静态的重定向方法就显得力不从心了。能否找到更有效的请求路由方法已成为 CDN 服务满足未来新应用需求的一个门槛。Anycast 请求路由为解决异构平台的内容分发提供了较好的解决思路。目前已有一些利用 Anycast 原理来对距离最近和负载最轻服务器进行定位的方案^[11,18]。

(3)实现精细化控制。现在的请求路由方法主要根据网络或服务器状况定位最佳服务节点,以此引导用户请求。随着用户需求的复杂化,在请求路由过程中也应该考虑用户,将用户需求也作为资源定位算法的度量准则之一。进一步地,还可以考虑内容提供商的因素,如所涉及的费用问题等。综合多种度量准则选择最佳服务节点,是未来实现更加精细的内容管理及重定向的一个发展趋势。

(4)有效控制成本。除考虑技术本身所需的开发和部署成本外,还要考虑各种系统管理成本。CDN 向云计算模式发展的一个重要特点在于其商业模式能够使用户以 pay-as-you-go 的方式分发内容,这不仅能为用户带来利益,还能使 CDN 提供商通过按使用收费的模式配置资源,避免大量基础设施的投资。相应地,请求路由作为 CDN 的一项关键技术,在成本控制方面也需要更好的解决方案。

结束语 请求路由问题是 CDN 的一个核心问题,一直以来都是学术界和工业界的研究热点。它一方面继承了一些传统路由技术的思想,另一方面又结合 CDN 的特点,实现了在 CDN 中对内容资源定位和对用户请求的重定向。随着网络和通信技术的发展,以及各种新应用的出现,现有的请求路由方法将面临新的挑战。本文从 CDN 请求路由的含义入手,从资源定位和请求重定向两个角度总结了请求路由的各种方法及存在的问题,以及相关研究的发展历程,并在此基础上,探讨了在当前网络环境和各种新应用的驱动下,请求路由研究存在的挑战和可能的发展方向。

参考文献

- [1] Cardellini V, Colajanni M, Yu P S. Request redirection algorithms for distributed Web systems [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2003, 14(4): 355-368
- [2] Yin Hao, Liu Xue-ning, Min Geyong, et al. Content delivery networks: a bridge between emerging applications and future IP networks [J]. *IEEE Network*, 2010, 24(4): 52-56
- [3] Pathan A M K, Buyya R. A taxonomy and survey of content delivery networks [D]. Parkville, Australia: Grid Computing and Distributed Systems (GRIDS) Laboratory Department of Computer Science and Software Engineering, University of Melbourne, 2006
- [4] Sivasubramanian S, Szymaniak M, Pierre G, et al. Replication for web hosting systems [J]. *ACM Computing Surveys (CSUR)*, 2004, 36(3): 291-334
- [5] Mah B A. An empirical model of HTTP network traffic [C]// *Proc. IEEE Annual Joint Conference of the IEEE Computer and Communications Societies*, 2002, 2: 592-600
- [6] Wang Li-min, Pai V, Peterson L. The effectiveness of request redirection on CDN robustness [J]. *ACM SIGOPS Operating Systems Review*, 2002, 36(SI): 345-360
- [7] Rabinovich M, Aggarwal A. RaDaR: A scalable architecture for a global Web hosting service [J]. *Computer Networks*, 1999, 31(11-16): 1545-1561
- [8] Kabir M H, Manning E G, Shoja G C. Request-routing trends and techniques in content distribution network [C]// *Proc. International Conference on Computer and Information Technology (ICCIIT)*, 2002
- [9] Barbir A, Cain B, Nair R, et al. RFC3568: Known content network (CN) request-routing mechanisms [S]. United States: RFC Editor, 2003
- [10] Szymaniak M, Pierre G, Steen M. Netairt: A DNS-based Redirection System for Apache [C]// *Proc. International Conference WWW/Internet*, 2003
- [11] Al-Qudah Z, Lee S, Rabinovich M, et al. Anycast-aware transport for content delivery networks [C]// *Proc. the 18th International Conference on World Wide Web*, 2009: 301-310
- [12] Guo Chuan-xiong, Lu Guo-han, Li Dan, et al. BCube: A high performance, server-centric network architecture for modular data centers [J]. *ACM SIGCOMM Computer Communication Review*, 2009, 39(4): 63-74
- [13] Rabinovich M, Rabinovich I, Rajaraman R, et al. A dynamic object replication and migration protocol for an internet hosting service [C]// *Proc. the 19th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2002: 101-113
- [14] Delgadillo K. Cisco Distributed Director [R]. Cisco White Paper, Cisco Systems, Inc., 1999
- [15] DeCandia G, Hastorun D, Jampani M, et al. Dynamo: amazon's highly available key-value store [J]. *ACM SIGOPS Operating Systems Review*, 2007, 41(6): 205-220
- [16] Kangasharju J, Ross K W, Roberts J W. Performance evaluation of redirection schemes in content distribution networks [J]. *Computer Communications*, 2001, 24(2): 207-214
- [17] Rodriguez P, Kirpal A, Biersack E W. Parallel-access for mirror sites in the internet [C]// *Proc. IEEE INFOCOM*, 2000, 2: 864-873
- [18] Alzoubi H A, Lee S, Rabinovich M, et al. Anycast cdns revisited [C]// *Proc. the 17th International Conference on World Wide Web*, 2008: 277-286
- [19] Jernberg J, Vlassov V, Ghodsi A, et al. Doh: A content delivery peer-to-peer network [J]. *Parallel Processing*, 2006: 1026-1039
- [20] Andrews M, Shepherd B, Srinivasan A, et al. Clustering and Server Selection Using Passive Monitoring [C]// *Proc. IEEE INFOCOM*, 2002, 3: 1717-1725
- [21] Ardaiz O, Freitag F, Navarro L. Improving the Service Time of Web Clients Using Server Redirection [J]. *ACM SIGMETRICS Performance Evaluation Review*, 2001, 29(2): 39-44
- [22] Liu Xue-ning, Yin Hao, Lin Chuang. A Novel and High-quality Measurement Study of Commercial CDN-P2P Live Streaming [C]// *Proc. International Conference on Communications and Mobile Computing*, 2009, 3: 325-329
- [23] Ball N, Pietzuch P. Distributed content delivery using load-aware network coordinates [C]// *Proc. ACM CoNEXT Conference*, 2008: 1-6
- [24] Armbrust M, Fox A, Griffith R, et al. Above the clouds: A Berkeley view of cloud computing [R]. Berkeley: EECS Department, University of California, 2009
- [25] Pierre G, Van Steen M. Globule: a collaborative content delivery network [J]. *IEEE Communications Magazine*, 2006, 44(8): 127-133
- [26] Bakiras S. Approximate server selection algorithms in content distribution networks [J]. *IEEE International Conference on Communications*, 2005, 3: 1490-1494
- [27] Wen Zhi-hua, Rabinovich M. Network distance estimation with dynamic landmark triangles [C]// *Proc. ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, 2008: 433-434
- [28] Nygren E, Sitaraman R K, Sun J. The Akamai Network: A platform for high-performance Internet applications [J]. *ACM SIGOPS Operating Systems Review*, 2010, 44(3): 2-19
- [29] Ballintijn G, Van Steen M. Characterizing Internet performance to support wide-area application development [J]. *ACM SIGOPS Operating Systems Review*, 2000, 34(4): 41-47
- [30] Vakali A, Pallis G. Content delivery networks: status and trends [J]. *IEEE Internet Computing*, 2003, 7(6): 68-74
- [31] Masa M, Parravicini E. Impact of request routing algorithms on the delivery performance of content delivery networks [C]// *Proc. IEEE International Conference on Performance, Computing, and Communications*, 2003: 5-12
- [32] Rodriguez P, Sibal S. SPREAD: Scalable platform for reliable and efficient automated distribution [J]. *Computer Networks*, 2000, 33(1-6): 33-49
- [33] Pierre G, Van Steen M, Tanenbaum A S. Dynamically selecting optimal distribution strategies for Web documents [J]. *IEEE Transactions on Computers*, 2002, 51(6): 637-651
- [34] KaZaa 官方网站 [EB/OL]. <http://www.kazaa.com/>
- [35] Pallis G, Vakali A. Insight and perspectives for content delivery networks [J]. *Communications of the ACM*, 2006, 49(1): 106
- [36] Heddaya A, Mirdad A. WebWave: Globally load balanced fully distributed caching of hot published documents [C]// *Proc. the 17th International Conference on Distributed Computing Systems*, 2002: 160-168

读取标签的 ID 号,然后单片机再读取电子罗盘的方向数据,并把这些数据封装成数据包,通过 ARM9 的串口,发送给 ARM9 主板。

第三步 数据解包、计算定位坐标。ARM9 把接收到的数据包解装,然后查看 ID-坐标映射表,把 RFID 号替换成坐标,再通过定位算法(前面已具体介绍过),计算出车载监控系统的中心位置坐标。

第四步 发送定位数据至网络。ARM 把车载监控系统的中心坐标和前进方向的定位数据通过网络发送给远程系统,并显示车载监控系统的定位图。

如此不断地重复第二至第四步,直至关闭车载监控系统。

结束语 本文提出的基于 RFID 的室内定位方法具有良好的移植性和可扩展性。在移植到不同大小房间时,本方法只需修改 ID-坐标映射表,而无需更改任何硬件设备。基于此定位技术,实现了一个“支持 Internet 远控化的可视智能室内车载监控系统”,验证了算法的有效性。

在未来工作中,将进一步研究读卡器大小及读卡范围对定位精度的影响,优化系统性能。

参 考 文 献

[1] 王志浩. RFID 定位的室内机器人管理系统[D]. 南台科技大学

(上接第 7 页)

[37] 严文中. CDN 负载均衡和内容路由研究 [D]. 杭州:浙江大学, 2005

[38] Hofmann M, Beaumont L R. Content networking: architecture, protocols, and practice [M]. San Francisco, CA, USA; Morgan Kaufmann Publishers, 2005

[39] Nortel Networks. Web OS Switch Software 1 0 . 0 Application Guide [EB/OL]. <http://sysdoc.doors.ch/NORTEL/212777-A.pdf>

[40] Foundry Networks. Global Server Load Balancing with Serveriron [EB/OL]. <http://www.genesisglobalinc.com/PDF/foundry-global-server-load-bal.pdf>

[41] Sivasubramanian S, Szymaniak M, Pierre G, et al. Replication of Web Hosting Systems [J]. ACM Computing Surveys, 2004, 36 (3): 291-334

[42] Weber S, Cheng Liang. A survey of anycast in IPv6 networks [J]. IEEE Communications Magazine, 2005, 42(1): 127-132

[43] 张丽, 严伟, 李晓明. Anycast——IP 的又一通信模式 [J]. 计算机研究与发展, 2003, 40(6): 784-790

[44] Fei Z M, Bhattacharjee S, Zegura E W, et al. A Novel Server Selection Technique for Improving the Response Time of a Replicated Service [C]//Proc. the 17th Annual Joint Conference of the IEEE Computer and Communications Societies. 2002, 2: 783-791

[45] Katabi D, Wroclawski J. A framework for scalable global IP-anycast(GIA) [J]. ACM SIGCOMM Computer Communication Review, 2001, 31(2 supplement): 186-219

资讯工程所, 2009

[2] 吴建中, 林毓棠, 陈怡云, 等. 透过 RFID 定位的寻宝机器人 [EB/OL]. <http://eiral.csie.stut.edu.tw/node/118>, 2009-12-08

[3] 游战清, 李苏剑. 无线射频识别技术(RFID)理论与应用[M]. 北京: 电子工业出版社, 2004

[4] 秦军, 龙建军, 吴百海. 一种超声波三维定位系统的研究[J]. 机床与液压, 2008, 36(8): 209-215

[5] 张东军, 郭文成, 刘莲花. 基于红外超声波光电编码器的室内移动小车定位系统[J]. 微型电脑应用, 2007, 23(5): 1-3

[6] 郑静, 张杰, 华泽钊, 等. 室内超声波定位系统的构建与测试[J]. 上海理工大学学报, 2007, 29(6): 553-561

[7] 罗超. 蓝牙定位测量[J]. 海峡科学, 2007(10): 56-57

[8] Chen Y, Francisco J A, Trappe W, et al. A Practical Approach to Landmark Deployment for Indoor Localization[C]//IEEE Communications Society subject matter experts for publication in the IEEE SECON proceedings. 2006, 1: 365-373

[9] Harter A, Hopper A. A Distributed Location System for the Active Office[EB/OL]. IEEE Network, January 1994

[10] Bahl P, Padmanabhan V N. Radar: An in-building rf-based user location and tracking system[C]//Proceedings of the IEEE International Conference on Computer Communications (INFOCOM), March 2000

[46] Gritter M, Cheriton D R. An architecture for content routing support in the internet [C]//Proc. the 3rd conference on USENIX Symposium on Internet Technologies and Systems. 2001, 3: 4

[47] Pan Jian-ping, Hou Y T, Li Bo. An overview of DNS-based server selections in content distribution networks [J]. Computer Networks, 2003, 43(6): 695-711

[48] Pathan A M K, Broberg J A, Bubendorfer K, et al. An architecture for virtual organization(VO)-based effective peering of content delivery networks [C]//Proc. the 16th IEEE International Symposium on High Performance Distributed Computing (HPDC). 2007: 29-38

[49] Thaler D G, Ravishankar C V. Using name-based mappings to increase hit rates [J]. IEEE/ACM Transactions on Networking, 2002, 6(1): 1-14

[50] Kapadia A, Triandopoulos N. Halo: High-assurance locate for distributed hash tables [C]//Proc of NDSS. 2008

[51] Saroiu S, Gummadi K P, Gribble S D. Measuring and analyzing the characteristics of Napster and Gnutella hosts [J]. Multimedia systems, 2003, 9(2): 170-184

[52] Clip2 Distributed Search Solutions. The Gnutella Protocol Specification v0. 4 [EB/OL]. <http://www.content-networking.com/papers/gnutella-protocol-04.pdf>

[53] Fujita Y, Yoshida J, Yoshida K, et al. Network Information Mining for Content Delivery Route Control in P2P Network [C]//Knowledge-Based Intelligent Information and Engineering Systems. 2004: 516-522