

基于负载均衡的分布式 IMS 体系架构的优化

罗 威¹,姜元建¹,李 洋¹,郎 赫²,张宇辰²,王慧敏³

- (1. 南瑞集团公司 国网电力科学研究院,江苏 南京 211000;
2. 国网天津市电力公司信息通信公司,天津 300010;
3. 南京邮电大学 通信与信息工程学院,江苏 南京 210003)

摘 要:IP 多媒体子系统(IMS)作为通用平台能够快速、标准化地创建移动服务。IMS 典型的部署模型支持集中的会话控制和应用程序服务器。而且,它们依靠复杂的边界元素来完成功能服务,例如政策执行。一个简单构架的第一步是将会话控制、应用服务和多媒体功能融合在一种边缘元素中。这种分布式的 IMS 体系拥有相同的处理节点,大大降低了系统的复杂性和功能模块实例化的规模。但是,分布式架构会减少统计增益,因此提出了一种负载均衡概念来解决这个问题。该机制在较小的控制和消息开销的基础上实现了一个大的虚拟的集中式服务器。该负载分享机制是轻量的,且开销最小。可以根据估计负载分享参数的方程进行微调,这个方程的输入包括重定位的开销和负载不对称参数 x 。仿真结果表明了该算法的基本特征,并且阐述了负载均衡的益处,能够为进一步完善负载分享的具体实施提供帮助。

关键词:移动服务平台;IP 多媒体子系统构架;分布式;负载均衡

中图分类号:TN929.5

文献标识码:A

文章编号:1673-629X(2017)11-0187-05

doi:10.3969/j.issn.1673-629X.2017.11.040

Optimization of Distributed IMS Architecture Based on Load Balancing

LUO Wei¹,JIANG Yuan-jian¹,LI Yang¹,LANG He²,ZHANG Yu-chen²,WANG Hui-min³

- (1. State Grid Electric Power Research Institute of NARI Group Corporation,Nanjing 211000,China;
2. Information Communication Company of State Grid Tianjin Electric Power Company,Tianjin 300010,China;
3. Institute of Telecommunications & Information Engineering,Nanjing University of Posts and Telecommunications,Nanjing 210003,China)

Abstract:The IP Multimedia Subsystem (IMS) serves as universal platform for fast and standardized creation of mobile services. Typical deployment models for the IMS favor centralized session control and application servers. Furthermore,they rely on sophisticated border elements for function service like policy enforcement. One step towards a simpler architecture is the integration of call control,application server and media functions into border elements,which leads to a distributed IMS architecture with equal processing nodes and drastically reduces system complexity and scales on functional module instantiation basis. However,it reduces the statistical gain. Therefore,a concept of load balancing is proposed to solve the problem. The load-sharing mechanism is lightweight and minimal overhead and can be fine-tuned based on an equation that estimates the load-sharing parameters,including the relocation overhead and the load asymmetry parameter x . The simulation results show its basic characteristics and discuss the benefits of load balancing,which will be helpful for further improving the implementation of load sharing.

Key words:MSP;IMS;distributed;load balancing

0 引 言

基于 IP 技术的下一代网络的发展促进了公共语音电话的发展^[1]。3GPP IP 多媒体子系统(IMS)及其扩展 ETSI TISPA 为 NGN 的全 IP 网络端到端多媒体会话业务的融合会话控制功能框架提供了标准^[2]。这

些下一代框架不仅仅旨在代替现有的网的大小来获得相应的资源。节点只有在高负载的情况下会卸载会话。负载系数的定义对高负载具有一定影响。

负载系数是基于负载等级 ρ 计算出来的。负载等级是前一个测量时间段里的已用资源与可用资源的比

收稿日期:2016-11-14

修回日期:2017-03-07

网络出版时间:2017-07-19

基金项目:国家自然科学基金资助项目(61302100);国网天津市电力 2016 年科技项目(KJ16-1-16)

作者简介:罗 威(1982-),男,工程师,研究方向为 IMS 技术研究及软件开发。

网络出版地址:<http://www.cnki.net/kcms/detail/61.1450.tp.20170719.1110.050.html>

值。 ρ 只定义了分配给每一个用户通话的 IMX 功能实体的资源占用率。所以,当 $\rho = 0.0$ 时,用户只是完成了 PSTN,而且想为快速的服务创新提供一个平台。IMS 定义了会话控制、应用服务和多媒体传输等功能,实现可以是分布式的,也可以是集中式的^[3-4]。在大多数配置场景中,会话边缘控制是放置在网络边界处,传统的 SBC 只执行会话控制函数的一部分。文献[5]提出了一个 IP 多媒体交换的概念,是一个融合了会话控制、应用服务和边界多媒体的可替换架构,减少了集中式的部件。这个平台的边缘包括了实现 IMS 功能的目的节点。这样的融合不仅简化了平台的管理和操作,而且减少了系统内部信令的复杂度。但是,该架构也带来了一定的系统开销,由于不可预测的交通流量、突发流的影响和增值业务(电话投票、平台失败),有些分布式节点会出现过载的情况。所以,会话可能在某个节点丢失,而此时不妨存在其他仍有很多空闲空间的节点。理论上,可以通过度量每个节点最坏的情况来避免这样的情况,但是开销太大。相对于集中式设计来说太浪费资源了,因此分布式设计需要负载均衡机制在不同的节点上进行分发处理^[6]。负载分享不是一个新的概念,已经在 PSTN 设备^[7]和大规模的服务集群上广泛使用^[8]。IMS 服务器的局部负载均衡机制已有研究^[9],过载信令也在不断的研究改进中^[10]。但是在高分布的 IMS 系统中进行实现却遇到了一些困难。

因此,文中系统性分析了在此环境中的负载分享机制,提出了负载分享的设计方法和实现,并进一步提出一种可扩展的负载平衡机制,以及一种轻量级的重定位机制。

1 分布式 IMS

1.1 IMS 部署的自由度

IMS 定义了许多不同的 CSCFs 和多媒体资源功能 MRF,服务逻辑在 AS 中实现。P-CSCF 位于接入网和核心网的边缘,以实现安全措施和用户注册功能。I-CSCF 位于网络之间。P-CSCF 和 I-CSCF 都在实体会话边缘控制实体 SBCs 中的映射实现。为了加强边缘元素的概念,一个更加合理的融合方案 IMX 被提出。IMX 不仅实现了 P-CSCF 和 I-CSCF,而且 S-CSCF 和 AS 以及多媒体能力功能也包含在此实体中。IMX 概念代替了集中式服务器,集中服务器利用几个认证节点处理所有用户的需求。IMS 平台包括几个分布式的 IMX 节点。IMX 最大的优势在于在 CSCFs 和 AS 之间的 SIP 信令通信是通过优化的中间件实现的,是集群内的通信,这个中间件也起到了传输和接入会话和用户信息的作用,同时对外集群提供标准的 IMS 接口。

利用 IMX 能够实现混合配置,即平台的一部分是由 IMX 簇实现,而另一部分是由 IMS 部件实现。另一个优势在于,IMX 节点能够充分利用先进的应用服务技术^[11]。

1.2 负载分享的优势

典型的下一代网络平台的配置是关系紧密的服务服务器作为一个簇,放置在一个地方,簇越少越好^[12]。负载均衡机制是由一个簇将接入网传来的会话进行分布处理,使得簇中的每个服务器负载平等,进而提高利用率,如图 1 所示。

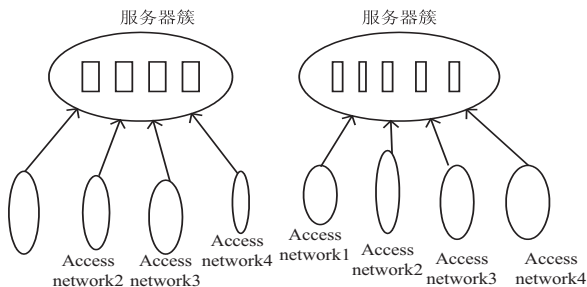


图 1 集中式负载均衡网络

在这种场景中,一个服务器簇处理多个接入网会产生统计增益。而在 IMX 中,资源的处理不是集中式的,而是由许多边缘节点分布处理的。如果每个 IMX 节点只处理与直接相连的接入网中的会话,统计增益就会大大减少,如图 2 所示。

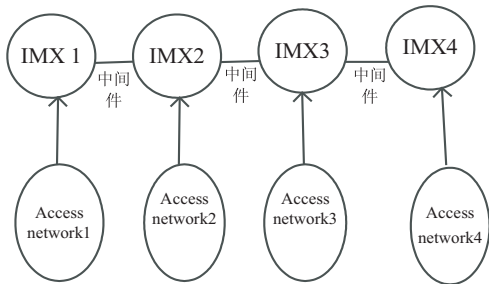


图 2 分布式负载均衡网络

为了防止过载,负载均衡技术可以将待处理的负载从高负载节点移动到低负载节点^[13]。另外,分布式的负载均衡技术可以获得比集中式更好的统计增益。但是,典型的服务器簇里的负载均衡技术和 IMX 中的负载均衡技术不同:前者是由集中式的调度器将用户分配到服务器,以实现均衡的负载分布;后者的目标不是均衡的负载分布,而是在过载情况发生之前进行互动协调。另外,已经处理完负载的 IMX 节点仍然在用户和真正处理负载的节点之间的路径上。

2 IMS 中的负载分享机制

负载均衡机制是由负载信息收集机制提供的负载情况来分配任务的。将负载信息收集进一步划分为测量和更新机制,将资源分配进一步划分为传输、定位和

重定频率,形成局部节点的负载。更新机制决定怎样分享机制,如图 3 所示。

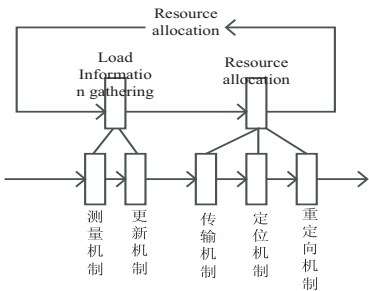


图 3 负载均衡机制

测量机制定义了系统执行负载测量的配负载信息。基于全局负载的角度,传输机制具体说明了什么时候重分配或者重定位任务。当重分配或者重定位被触发时,定位机制决定负载是如何在节点之间转移的,重定向机制决定有多少负载要被重定位。

为了获得全局负载信息和 IMX 的任务分配,节点之间通过通信实现状态更新,IMX 的更新机制采取的是集中式的、基于时间的、拉模型的。而分配任务机制采取的是分布式的、基于值的、推模型的。

典型的服务器簇的设计是在负载均衡器的后面有许多服务器。IMX 节点不同于服务器簇,即 IMX 节点总是在路径上负责处理一些任务。网络服务器之间的转换是通过直接在栈里重写地址实现的。但是,由于 IMX 节点的重定位的花销很大,所以这种静态的固定速率需求的资源分配方式不适合 IMS。

总之,目前没有方法适合 IMX 场景,所以文中提出一种合适的方法——分布式的 IMS 平台的负载均衡机制,即包括重定位和资源管理的负载均衡机制。

IMX 能够动态创建和分配功能实体,而这些实体是由局部资源管理器(LRM)进行管理的,这些实体和 LRM 都在 IMX 节点中^[14]。当创建这些实体时,有内存但是没有 CPU 资源,因为还没有被分配给用户,所以是无状态和没有被激活的。

IMX 处理用户注册和会话如图 4 所示。

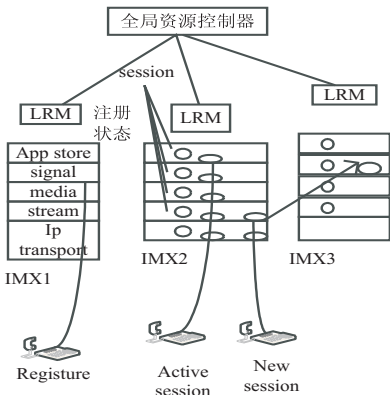


图 4 会话重定位

IMX1 节点中的功能实体包括可重定向的和非可重定向的部分,入口 IMX 节点会执行非可重定向功能。在用户执行非可重定向功能如注册功能时,IMX 中的调度员会给用户配备一个用户控制管理器(UCM)。根据用户的行为,UCM 请求 LRM 分配相应的功能实体给所属的用户,并控制着合适的功能链和这些功能实体的会话状态。如果一个新的会话被初始化后发现入口节点的资源很少,UCM 不会从 LRM 中获得所需的功能实体,而是由 LRM 触发重定位功能,即 UCM 会将远处节点的功能实体分配到该用户并传输所需的状态信息,继而业务也被重定向。入口节点会对用户保持接触点并处理注册功能。新会话到达 IMX2 后被重定位到 IMX3,但是用户的注册仍然是由 IMX2 节点完成。IMX2 中现存的会话将不会被重定位。

2.1 负载信息的收集和更新

在上面的负载分享机制中,每一个 IMX 节点独立决定是否采取重定位策略,所以这种资源分配机制是一种基于值的分布式机制。这使得节点能在高负载情况下快速做出反应以及会话到达时能够及时进行重定位。

与分布式的资源分配机制不同,负载信息的收集和更新机制采用集中式的机制来决定何时对节点中的负载信息做出反应。为减少 IMX 节点需要处理的负载更新消息的数量,网络采用星型拓扑结构,其中全局资源控制器(Global Resource Manager, GRM)是中心节点。为了获得及时消息,采用基于时间的策略来周期性触发负载消息的更新。由于 IMX 平台是基于高带宽的核心网,所以在 IMX 节点之间传播消息是无需吝啬的。基于三向握手的“拉模型”比较适合这种触发过程,原因在于这种“拉模型”更新快并且能给所有的 IMX 节点提供新消息。

图 5 描述了 IMX 更新机制。

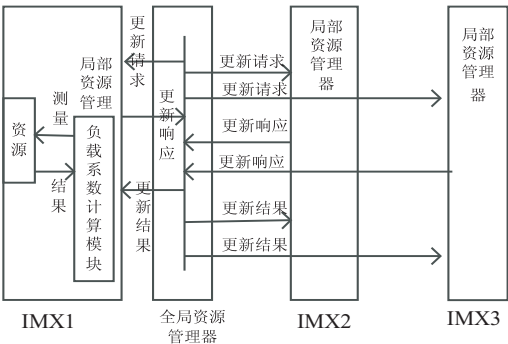


图 5 测量和更新机制

GRM 通过发送 updateRequest 消息给所有 IMX 节点的 LRM 来触发更新。IMX 节点在接收到这条消息后,测量节点内的平均资源利用率,并把它转化成负载

参数(在下一节中具体介绍)。每个 IMX 节点发送包含负载参数的更新回复消息给全局资源控制器。一旦全局资源控制器接收到更新回复消息,它会给所有的 IMX 节点发送一个包含所有节点负载消息的更新结果消息。在这个机制下,所有的负载情况会在很短的时间里到达全局资源控制器,然后被直接包含在更新结果消息中。

根据消息和数据速率,更新机制的开销可以被量化。假如有 n 个 IMX 节点,每个节点在更新周期 μ 内有 3 个消息需要发送给全局资源控制器,即在全局资源控制器中接收的消息速率为:

$$m = \frac{3n}{\mu} \quad (1)$$

假设一个更新请求包或者更新回复包的大小为 L ,更新结果包的大小为 nL ,对于全局资源控制器来说,更新机制的数据速率为:

$$r = \frac{2nL + n * n * L}{\mu} = \frac{L}{\mu}(2n + n^2) \quad (2)$$

假设 $\mu = 1$ s, $L = 50$ B, $n = 50$ 。根据式(2)可得数据速率为 1.04 Mbps,每秒传输 150 条消息。与 SIP 协议相比,IMX 节点的更新协议上的消息和数据速率可以忽略不计。

2.2 资源分配

因为会话不能被完全卸载,而且重定位的开销很大,所以所有会话应该尽可能在入口的节点中被处理。因此,早期的会话重定位不能获得一个均衡的负载分布。而且,IMX 平台应该根据它们所服务的接入注册,并没有会话被激活。当 $\rho = 1.0$ 时,IMX 节点的资源只能处理现有的会话,新来的会话已经无法处理,此时的超负荷控制机制已经被激活了。

文中的重定位机制将负载系数分成三类。当 $0 \leq \rho \leq \varphi$,节点处于可以接收会话的状态;当 $\varphi \leq \rho \leq \omega$,节点处于激活状态;当 $\omega \leq \rho \leq 1.0$,节点处于过载状态。在负载分享机制中,每一个 IMX 节点在内部定义了如何根据当前的负载系数重定位会话,是一种分布的、基于值的、源驱动的机制。图 4 中的传输机制是指,只要 IMX 节点在超载的状态下,就会卸载会话;定位机制则具体说明了会话卸载将会转移到哪个处于接收状态的节点上,处于过载状态的节点以循环方式从 updateResult 消息中处于接收状态的节点中选择。在重定位机制中,过载节点会将所有的新会话卸载。

负载分享机制在 IMX 节点上只增加了测量和更新机制,参数 φ 、 ω 都可以根据负载情况和系统参数事先计算出来。过载状态可以通过转移负载切换到接收状态,中间有一个激活状态。激活状态是滞后系统的因素,增加激活状态的范围会减少处于接收状态的

节点,就会限制系统的整体容量。如何设置激活状态的门限在很大程度上取决于系统实现和对于震荡的敏感度,所以文中不考虑这个阶段。

3 负载分享的定量性分析

在负载均衡机制中考虑两个节点的场景,首先计算出静态环境下的最佳临界值,再通过随机负载的仿真评估这个系统。

3.1 系统模型

为了研究 IMX 系统的基本功能,从实现和配置的具体细节中抽象出一个系统模型,包含体系构架、资源分配和加载在系统上以通信量为模型的负载。

网络平台中有几个接入网络,每个接入网络连接着分布在平台边缘的 IMX 节点,IMX 节点之间由高速网络互连。所以,网络影响在负载分享机制中可以忽略。假设 IMX 节点有合适的过载控制机制,即当容量过载时会放弃新需求。节点的负载等级取决于资源容量和已消耗的资源。

仿真中,假设会话的持续时间是平均值为 h 的指数分布,会话的到达率 λ_i 服从指数分布且平均值取决于接入网中的负载。这两个假设对于会话流来说都是合理的。因此,加载在 IMX i 上的业务负载为 $A_i = \lambda_i * h$, n 个 IMX 节点的平均负载为 $A = \frac{1}{n} \sum_{i=1}^n A_i$ 。为了研究负载均衡技术,引入参数 x 来描述负载不对称的情况。假设一半的接入网络是高负载,即

$$A_1 = (1 + x)A \quad (3)$$

另一半的接入网络是低负载,即

$$A_2 = (1 - x)A \quad (4)$$

一次会话所消耗的资源取决于当地的或者是重定位的开销 e 和额外的重定位的开销 f 。 F 包括 SIP 解析、加密、解密和重定位时在节点 i 中的传输开销。重定位的开销率为:

$$b = 1 + \frac{f}{c} \quad (5)$$

假设已经激活的会话的资源开销是常量。通常同一时间会有几百个已经被激活的会话。在该模型中,会话只影响一个 IMX 节点,符合增值的服务会话情况。在两个客户之间的端到端的会话对应于两个服务会话,表 1 给出了模型参数和相应的值。

3.2 理想仿真环境

评估需要考虑在已知两个 IMX 节点的 x 以及 b 的情况下,如何设定负载分享参数 w 。正如前面讨论的,不考虑中间的“激动状态”($\varphi = \omega$),但是在计算时考虑 φ 是很直观的。所有节点都采用相同的 ω ,为了评估出最合适的 ω ,考虑静态环境,即会话持续时

间是无穷大的。图 5 中有 2 个 IMX 节点,而且节点 1 的负载是 A_1 ,节点 2 的负载是 A_2 , $A_1 > A_2$ 。负载 l_i 在本地处理,而 r_i 被重定向处理,即

$$A_i = l_i + r_i$$

(6)

表 1 模型参数

符号	描述	默认值
A	归一化系统负载	无,输入参数
h	平均会话持续时间	180 s
λ_i	平均会话建立速率	$\frac{A_i}{h}$
c	IMX 节点的资源处理能力	1 000 000
e	处理会话的 IMX 节点上的资源开销	1000
f	会话重定位节点 i 上每一个会话的资源开销	50
b	重定位开销率	1.05 (由 e 和 f 决定)
x	负载不均衡系数	0.5

图 5 表明了理想情况下的资源分配,节点 1 中的所有资源都被利用了,而节点 2 正好到达了可接收和过载的临界值。节点 1 本地处理了会话 l_1 ,将会话 r_1 重定位到节点 2;而节点 2 处理会话 l_2 。将 ω 设置高些会得到一个次优化的情况,因为会导致节点 1 重定向过晚,并且在节点 2 处于可接收状态时,节点 1 会处于阻塞。相反情况,当 ω 设置得低时,节点 2 过早地不处于可接收状态,节点 1 将会在过载状态下本地处理会话。即使是在理想状态下,当节点 1 发生阻塞时,节点 2 仍然有空闲的资源。从图 5 可以得到以下关系:

$$c = l_1 * e + r_1 * f$$

(7)

$$w \cdot c = l_2 \cdot e + l_1 \cdot e$$

(8)

$$l_1 = l_2 + r_1$$

(9)

由式(3)~(9)可以得到 ω 的最佳取值:

$$\omega = \frac{1}{1 - x + bx}$$

(10)

考虑到 x, b 的定义, ω 被限制在 $[0,1]$ 之间,根据表 1 可得 $\omega = 0.075$ 。

3.3 仿真环境

为了得出随机负载下的负载均衡机制的益处,利用事件驱动仿真库对 IMX 系统进行仿真。仿真环境允许会话的建立和节点之间信息交换时直接的负载更新,比如 IMX 和 GRM 之间的信息交换。接入网络作为节点模型,包括会话信息产生器,它是根据预定义 IAT 分布产生会话建立消息的,并且将会话持续时间作为消息参数加入到其中。一旦会话建立消息到达,IMX 节点会检查自身是否有足够的资源来处理会话。如果能在本地丢弃会话,根据当前的负载系数和节点在可接收状态的利用率,IMX 节点转发直接与其相连的接入网产生的会话消息到其他的 IMX 节点。根据负载更新协议,LRM 和 GRM 交换负载更新信息。对于每一个参数设置,做十次仿真,每次会话是在 1 百万到 1 千万个方向数据

3.4 仿真结果

考虑了在负载不对称系数为常数 $X = 0.5$ 情况下的不同的负载值 A 。图 6 显示了在不同标准化负载的情况下,负载分享参数 ω 对阻塞概率的影响。

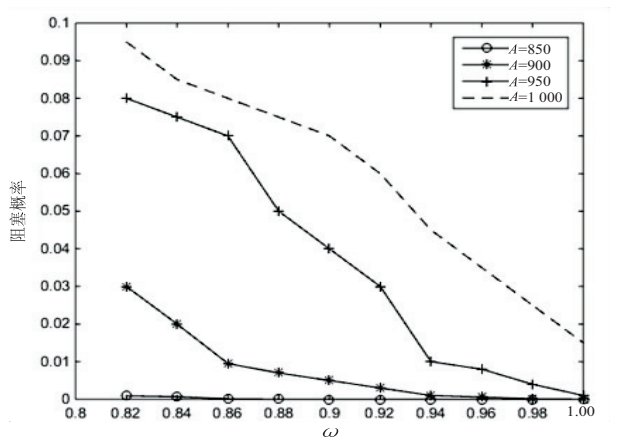


图 6 负载分享参数-阻塞概率关系曲线

假设有两个节点,对于某一个负载参数而言,都有一个最小的阻塞概率。为达到相同的阻塞概率,负载越高,负载分享参数越大。所以, ω 没有适用于所有情况的最优解。

如果 ω 设置得过高,IMX1 开始重定位会太晚,导致资源不够处理本地会话,而 IMX2 即使在过载状态下也能处理会话。如果 ω 设置得过低,IMX2 过早地离开可接收状态。因为 e 远远大于 f , ω 的轻微改动只影响一部分会话,所以将 ω 设置过低不会导致性能的大幅衰减,能够得到参数值设置的指导性原则。另外,理想静态情况下的估计是一个好的开头,微调必须要知道最大可容阻塞概率和期望业务负载。更多 IMX 节点的仿真得出了相似的结果。

4 结束语

分布式的 IMS 方法能够把所有的功能整合到边缘节点,它们配置简单却能提供较高的灵活性和可扩展性。但是在处理紧急负载改变时,用于在节点之间转移负载的负载分享机制是必不可少的。文中提出了一种分布式系统中的负载分享机制,分析了设计空间和被认可的一些机制。在此基础上,提出一种最适合分布式 IMS 实现需求的策略。该负载分享机制是轻量的,且开销最小,可以根据估计负载分享参数的方程进行微调。仿真结果显示了该算法的基本特征,并且阐述了负载均衡的益处。该研究为进一步完善负载分享的具体实施提供了基础和指导。

参考文献:

[1] 叶 涛. 基于软交换技术的下一代网络构架的研究[D]. (下转第 195 页)

网页是比较复杂的,那么差距将更加明显。

5 结束语

HTTP/2 的多路复用功能减少迟延,首部字段压缩降低了协议开销,请求优先级、服务器端推送更加增强了 HTTP/2 的优势^[19]。上述测试验证了 HTTP/2 可以很好地提高页面加载速度,为 Web 性能带来很大的提升,对改善用户体验有重要意义。文中只研究了使用 HTTP/2 后 Web 性能的提升,对使用 HTTP/2 的安全性能没有深入研究,特别是对 HTTP/2 没有强制要求使用 TLS,可能会带来一些安全性能方面的问题,这还有待进一步研究。

参考文献:

[1] 肖威华,赵 莺,陈 晖.影响用户 Web 浏览体验的因素分析[J].通信技术,2013(8):140-143.

[2] 孙志梅. B2C 电子商务网站用户体验评价研究[J]. 商,2015(26):78.

[3] 高克立. Web 性能的相关技术分析和研究[J]. 现代电信科技,2003(11):46-50.

[4] 李松峰. Web 性能权威指南[M]. 北京:人民邮电出版社,2014.

[5] Belshe M, Thomson M, Peon R. Hypertext transfer protocol version 2 (http/2) [S]. [s. l.]:[s. n.],2015.

[6] 王 靖. 面向 Web 加速的 HTTP 协议优化机制的研究与设计[D]. 北京:北京邮电大学,2013.

[7] 张宝玉. 浅析 HTTPS 协议的原理及应用[J]. 网络安全技术与应用,2016(7):36-37.

(上接第 191 页)

武汉:武汉理工大学,2006.

[2] 张 洁. 基于 IP 多媒体子系统应用于下一代网络的服务发现架构的研究[D]. 北京:北京邮电大学,2009.

[3] 王金柱. IMS 过载控制关键技术研究[D]. 北京:北京邮电大学,2012.

[4] 谭春光. 基于 IMS 和 P4P 的 IPTV 系统关键技术研究[D]. 沈阳:东北大学,2015.

[5] Wahl S, Oberle K, Kessler M, et al. The next step in IMS architecture - a comprehensive IMS network element[C]//Proceedings of broadband Europe 2006. Geneva, Switzerland; [s. n.],2006.

[6] 弭 伟. 基于 DHT 的分布式网络中负载均衡机制及其安全性的研究[D]. 北京:北京邮电大学,2012.

[7] Asif M, Majumdar S, Kopec G. Load sharing in call server clusters[J]. Computer Communications,2007,30(16):3027-3045.

[8] 伊万·里斯蒂奇. HTTPS 权威指南[M]. 北京:人民邮电出版社,2016.

[9] 高希然,翟金亭. HTTPS 探析[J]. 信息记录材料,2016,17(4):49-51.

[10] 吴维元,肖柳林,李荣辉,等. Web 服务数据传输通道的安全性分析[J]. 网络安全技术与应用,2008(2):85-86.

[11] Kihara B, Shimizu K. Considerations for protocols with compression over TLS (work in progress) [EB/OL]. 2017. <https://datatracker.ietf.org/doc/draft-kihara-compression-considered-harmful/>.

[12] 左 岸. HTTP2 问世可提升 Web 性能达 15% [J]. 通信世界,2015(22):36.

[13] Grigorik I. Making the web faster with HTTP 2.0[J]. Communications of the ACM,2013,56(12):42-49.

[14] Darwish N R, Abdelwahab I M. Impact of implementing HTTP/2 in web services [J]. International Journal of Computer Applications,2016,147(9):27-32.

[15] 范菁菁. HTTP/2.0 关键技术及标准化进展[J]. 电信网技术,2014(6):60-63.

[16] Peon R, Herve R. HPACK: header compression for HTTP/2 [S]. [s. l.]:[s. n.],2015.

[17] A simple performance comparison of HTTPS, SPDY and HTTP/2 [EB/OL]. 2015. <http://blog.httpwatch.com/2015/01/16/a-simple-performance-comparison-of-https-spdy-and-http2/comment-page-1/>.

[18] Adi E, Baig Z, Hingston P, et al. Distributed denial-of-service attacks against HTTP/2 services [J]. Cluster Computing, 2016,19(1):79-86.

[19] 潘建平,顾冠群,沈苏彬. Web/HTTP 协议性能分析和改进的研究[J]. 计算机研究与发展,1999,36(1):106-111.

[8] Cardellini V, Casalicchio E, Colajanni M, et al. The state of the art in locally distributed web-server systems [J]. ACM Computer Surveys,2002,34(2):263-311.

[9] Bessis T. Improving performance and reliability of an IMS network by co-locating IMS servers [J]. Bell Labs Technical Journal,2006,10(4):167-178.

[10] Hilt V, Widjaja I, Malas D, et al. Session Initiation Protocol (SIP) overload control [S]. [s. l.]:[s. n.],2014.

[11] 刘永佳. IMS 多媒体共享业务模式设计与实现[D]. 北京:北京邮电大学,2011.

[12] 王晓庆. 面向智能网演进的异构网络中间件体系研究[D]. 北京:北京邮电大学,2006.

[13] 王红斌. Web 服务器集群系统的自适应负载均衡调度策略研究[D]. 长春:吉林大学,2013.

[14] 赵 阳. 小型 IMS 应用系统的设计与实现[D]. 北京:北京邮电大学,2009.