基于 H.323协议的分布式网守研究

沈 伟 丁 嵘 张 辉

(北京航空航天大学 计算机学院, 北京 100083)

摘 要: 基于H.323协议的视讯会议系统的可扩展性与网守(GK, Gatekeeper)结构密切相关.提出了一种多区域分布式网守架构, 提供了多媒体视讯会议系统在大规模网络部署的方案.这种分布式结构不仅克服了集中式和分层式模型的缺点,还很好地解决了网守之间的负载平衡问题.终端之间的跨区域通信通过基于别名规则的信令消息路由方法来实现,网守通过扩散和逆向学习法自动生成信令消息散列表.考虑到网守之间的负载平衡问题, 本架构在区域内采用集中式管理模型,并提出了基于负载因子最小平衡算法(BLFA, Balancing Load Factor Algorithm)的网守负载平衡解决方案.

关 键 词: 视讯会议; 架构; 大规模系统; 网守; 负载平衡

中图分类号: TP 302.1

文献标识码: A 文章编号: 1001-5965(2007)01-0055-04

Study on H.323-based distributed gatekeepers

Shen Wei Ding Rong Zhang Hui

(School of Computer Science and Technology, Beijing University of Aeronautics and Astronautics, Beijing 100083, China)

Abstract: The scalability of H.323-based video conferencing system is closely related to the architecture of gatekeeper (GK). A distributed H.323 gatekeeper architecture based on multiple zones was proposed, providing the solution for the deployment of multimedia video conferencing system on the large scale network. This architecture did not only conquer the drawback of centralized architecture and hierarchical architecture, but could balance the load between gatekeepers. Design of GK-level signaling message routing based on alias format was used to solve the communication problem between endpoints registered in different zones. In the zone, the centralized architecture was adopted and the main-GK managed the slave-GKs. Balancing load factor algorithm (BLFA) was presented to balance the load between gatekeepers.

Key words: video conferencing; architecture; large scale system; gatekeeper; load-balancing

视讯通信产业已经成为IT业的新兴热点行业,越来越多的软件开发商投入到视讯会议系统市场的争夺中. H.323[1]是现在主流视讯会议系统所遵守的协议,它为分组网络的视讯会议系统定义了4个组件: 多点控制单元(MCU, Multiple Control Unit)、终端(terminal)、网关(gateway)和网守(GK, Gatekeeper). GK是整个H.323系统的管理者,它必须提供地址解析、接入控制等基本功能和呼叫路由、端点定位等可选功能. 然而, H.323协议是针对局域网提出的,因此在将基于H.323的视讯会议系统部署到大规模多区域网络时,其可扩展性就遇到了严峻的挑战,主要的瓶颈之一是GK的结构约束.

为了解决GK的可扩展性问题,国内外学者做了很多的研究工作,提出了多种结构模型. Radhika R. Roy提出过GK的集中式模型框架,并且罗列了实现这个结构的一些难点[2]. 郝又超、张晓等对分层GK结构模型作了较深入的研究,并提出了实现方案[3-4]. Li Rui等将Agent引入到GK结构中,其实质也是分层式的模型[5]. 无论是

收稿日期: 2006-02-23

基金项目: 中国下一代互联网(CNGI)视讯会议系统应用示范项目(CNGI-04-15-1A)

作者简介: 沈 伟(1981-), 男, 浙江湖州人, 硕士生, shenwe@nlsde.buaa.edu.cn

集中式模型,还是分层式模型,都遭遇到鲁棒性不佳的问题,某个关键 GK 的崩溃很可能会导致整个视讯会议系统的瘫痪.

RadhikaR. Roy还提出一种分布式 GK 的架构框架[6]. 该框架消除了系统对某些关键 GK 的依赖,每个 GK 都负责管理一个区域,其地位都是平等的,这能够解决集中式和分层式模型所带来的问题. 但是他的文章所描述的 GK 层信令消息路由方法有很大缺陷,存在着散列表膨胀问题,此外,他也没有考虑 GK 之间的负载平衡问题.

本文提出了一种具有较高消息路由效率和负载平衡能力的多区域分布式 GK 结构. 论文的结构是: 第 1章介绍分布式 GK 的结构框架下终端的呼叫过程,第 2章具体描述 GK 层的消息路由方法,平衡区域内 GK 负载的机制在第 3章给出,第 4章是结论.

1 分布式网守结构下终端呼叫过程

分布式 GK 的结构如图 1所示. 整个多媒体会议系统被分为多个区域,区域内采用集中式 GK 结构. GK_m 是每个区域唯一的主 GK, GK_s 是区域内的从 GK, GK_m 负责管理所在区域内的 GK_s,每个 GK_s 直接和 GK_m 通信, GK_s 之间不通信,区域中的终端最终注册在 GK_s 上. 区域间采用分布式结构,各个区域的地位是平等的,没有管理与被管理的关系. 区域间的通信通过各区域 GK_m 实现.

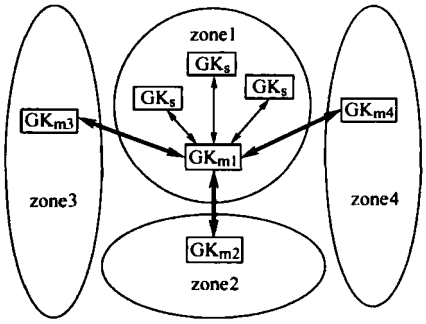

图 1 分布式网守结构

终端间的呼叫分为区域内呼叫和跨区域呼叫.

同在 zone1内的 E_1 向 E_2 发起的呼叫属于区域内呼叫,过程如图 2所示 (E_1 和 E_2 已分别在 GK_{s1}和 GK_{s2}上注册): E_1 向自己注册的 GK_{s1}发送呼叫请求 (ARQ, Admission Request)消息, GK_{s1}收到消息后发现被呼叫终端 E_2 并不在自己的注册终端列表中,就向 GK_{m1}转发定位请求 (LRQ, Location Request)消息, GK_{m1}收到消息后先在先前呼叫的缓存中查找是否有 E_2 的注册信息 (即注册在哪个 GK_s),若找到 (这里为 GK_{s2}),即向 GK_{s2}转发 LRQ消息,若找不到则向 zone1内除 GK_{s1}以外所有 GK_s 发送询问消息,在得到 GK_{s2}的答复消息后, GK_{m1}向 GK_{s2}转发 LRQ. GK_{s2}收到消息后发现 E_2 是自己的注册终端,就向 GK_{m1}发送定位确认 (LCF, Location Confirm)消息. GK_{m1}向 GK_{s1}转发这条 LCF消息,在收到 LCF后, GK_{s1}向请求终端 E_1 发送呼叫确认 (ACF, Admission Confirm)消息. 然后, E_1 发起呼叫建立信令,信令消息的发送过程与呼叫请求类似.

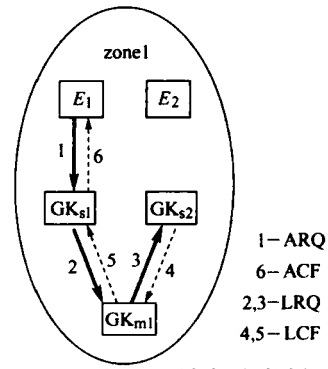

图 2 区域内呼叫过程

终端之间跨区域的呼叫需要 GK 层的消息路由机制. zone1中的 H. 323终端 E_1 呼叫 zone2中终端 E_2 的过程如图 3所示.

刚开始发生的事情和区域内呼叫的过程一样,也是从向注册的 GK_s 发送 ARQ 消息开始的,直到 GK_{m1}发现被呼叫终端 E_2 并不在 zone1内后,它向其邻居 GK_m (zone3内的 GK_{m3})转发 LRQ 消息, GK_{m3}发现被呼叫终端也不在其管理的 zone3中,再向其邻居 GK_m (zone2内的 GK_{m2})转发 LRQ 消息, GK_{m2}收到 LRQ消息,发现 E_2 已在自己管理的区域 zone2中注册,则向 GK_{m2}发送 LCF 消息. LCF消息沿着 LRQ 消息的相反路径返回至 E_1. 呼叫阶段的其他信令消息的发送过程与呼叫请求信令类似.

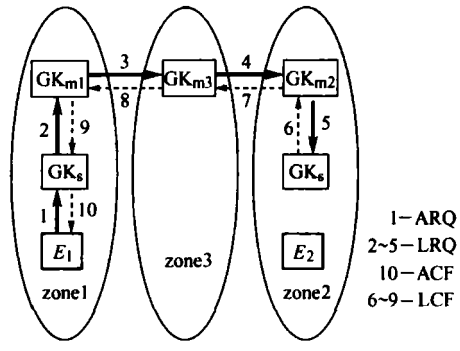

图 3 区域间呼叫过程

2　GK层信令消息路由

在上一章所提到的跨区域呼叫过程中, 若 GK_{m1}有多个邻居 GK_m, 将 LRQ 消息转发哪个 GK_m 要根据 GK_{m1}所维护的 GK 层的消息散列表而定. 在收到区域内 GK_s 转发过来的需要路由的信令消息后, GK_m 查找散列表, 找到有关被叫终端的散列后将消息转发至正确的邻居 GK_m.

2 1　GK 层消息散列表的自动生成

各个区域的 GK_m 可以通过扩散和逆向学习法来形成信息散列表. 对每一个需要路由的信令消息来说, 所有和它有关的 GK_m 可分为发送 GK_m、转发 GK_m 和目的 GK_m 三类. 下面仍以图 3 为例来说明信令消息散列表是如何通过扩散和逆向学习来形成的. 假设所有 GK_m 的消息散列表都是空的, GK_{m1}收到 LRQ 消息后, 发现被叫终端不在本区域内, 则搜索消息散列表, 没有找到至被叫终端的散列, 则 GK_{m1}将向全部的邻居 GK_m 转发这条 LRQ 消息. 这就是一个消息扩散的过程. 邻居之一 GK_{m3}在收到转发的 LRQ 消息后, 也没在所管理的区域中发现被叫终端, 然后同样是搜索自己的消息散列表, 也没有找到相关散列, 同样也进行消息扩散, 将 LRQ 消息转发到其全部邻居 GK_m; 邻居之一 GK_{m2}收到这个 LRQ 消息后, 发现呼叫请求的被叫终端已在本区域注册, 立即向转发这条信令消息的邻居 ——GK_{m3}回复定位确认消息 LCF. GK_{m3}收到这条 LCF 消息后, 分析和学习这对 LRQ和 LCF 消息, 提取有关信息, 生成到被叫终端的散列并添加到消息散列表中, 这就是逆向学习的过程. 处理完毕, GK_{m3}再将 LCF 消息转发给 GK_{m1}, GK_{m1}也同样地生成相应的消息散列. 如果将来经过它的信令消息中含有相同的目的终端, GK_{m3}就直接将消息转发给 GK_{m2}, 不再对全体邻居 GK_m 进行转发. 这个过程中, GK_{m1}是发送 GK_m, GK_{m3}是转发 GK_m, GK_{m2}是目的 GK_m. 路径上的每个 GK_m 都会对定位请求做相同的处理, 即转发或回复. LCF 消息将沿着定位请求消息的传送路径反向发送, 最后发送至发起呼叫终端区域内的 GK_m, 即发送 GK_m. 在消息传送过程中, 沿途路径上的转发 GK 会根据 LRQ 和相应的 LCF 消息将定位请求的目的终端和回复定位确认的邻居转发 GK 地址添加到消息表.

为了防止网络扩散 (flooding), 可以在信令消息中设置计数器字段, 消息每经过一个转发 GK_m, 计数器减 1, 当某个转发 GK_m 在收到的信令消息的计数器为 0时, 就将此消息丢弃. 当转发 GK_m 在收到信令消息后, 发现所管理的区域内没有目的终端, 并且除了发送这条消息的邻居转发 GK_m 外无其它邻居 GK_m, 也将这条信令消息做丢弃处理.

为了防止消息的重复处理和发送, 可以对每个消息进行编号, 确定唯一性, 当转发 GK_m 收到曾经转发过的消息后不再做任何处理. 此外, 发送 GK_m 需要设置一个计时器, 在消息发送的时候启动计时器, 如果当计时器超时之前未收到消息的回复, 可以选择重发消息或者认为目的终端不存在.

2 2　基于别名规则的消息散列表

按照 H. 323 协议规定, 目的终端的表现形式可以是 IP地址, 也可以是别名 (这时需要 GK 的地址翻译功能). 如果采用 IP地址代表目的终端, 则消息散列表散列的形式应类似为 〈目的终端 IP地址, 转发的相邻 GK_m 地址 〉. 相同区域内的终端在 IP地址上无任何相关性, 因此, 对于每个目的终端, 都需要生成一条散列. 这样 GK_m 维护的消息散列表是相当庞大的, 并且随着呼叫次数的增加, 散列表还会不断膨胀. 虽然可以根据一些算法选择缓存策略将散列表数量控制在一定范围内, 还是会对消息路由效率产生很大的影响.

本文提出的方法是用别名来表示终端, 并且别名必须符合一定的规则. 比较简单的规则是别名必须符合这样的形式: alias@ zonename 同一区域内终端别名的 zonename后缀必须是相同的, 这样来表征它们属于同一个区域. 这样, 同一区域内注册的终端别名就有了相关性, 即 zonename的后缀是相同的, 消息散列表的散列可以设计为〈zonename, 转发的相邻 GK_m 地址 〉, 到终端的散列变成到终端所在区域的散列, 相比之下, 消息散列表规模大大缩小. 有相同 zonename后缀的终端构成了一个区域, 区域不限定终端的地理位置和 IP地址. 这种方法的原理来自于网络层的路由, 它是根据目的 IP地址的网络地址来确定路由的, 而网络地址可以用掩码和 IP地址计算得出. 用区域的地址标识 (zonename后缀) 来代替网络地址, 就形成了 GK 层消息路由散列设计的基本思想.

一旦消息散列表形成之后, GK_m 在收到终端的信令消息时, 查看目的终端别名, 提取 zonename后缀, 搜索散列表找到将要转发的 GK_m 地址, 然后将信令消息转发出去. 这就是基于别名规则的 GK层消息路由过程.

3 区域内网守的负载平衡

本架构在区域内部采用集中式模型管理, GK_m 负责管理区域中 GK_s. GK_m 除了参与区域间的 GK 层消息路由之外, 还负责平衡区域内 GK_s 之间的负载. 所有 GK_s 和区域的 GK_m 相连, GK_s 之间不相互通信. 衡量 GK_s 的负载参数有两个, 分别为在 GK_s 上注册的终端数量 N_r 和 GK_s 所维护的通话路数 N_c. GK_m 必须实时掌握各个 GK_s 的负载情况, 才能对之后的终端注册和通话做出合理的分配, 做到平衡各 GK_s 间的负载.

GK_m 可以采用负载因子最小平衡算法来为 GK_s 分配终端注册和通话. 假设区域中有 GK_{s1}, GK_{s2}, ……, GK_{sn}, 有以下公式:

$$P_{ri} = \frac{N_{ri}}{N_{mri}} \qquad i \in (1, 2, \cdots, n) \qquad (1)$$

$$P_{ci} = \frac{N_{ci}}{M_{mci}} \qquad i \in (1, 2, \cdots, n) \qquad (2)$$

$$P_{rmin} = \min\{P_{ri}\} \qquad i \in (1, 2, \cdots, n) \qquad (3)$$

$$P_{cmin} = \min\{P_{ci}\} \qquad i \in (1, 2, \cdots, n) \qquad (4)$$

其中, P_{ri} 是 GK_{si} 的注册负载因子, $0 \leqslant P_{ri} < 1$, 数值越接近 1, 说明 GK_{si} 的注册负载越重. N_{ri} 是 GK_{si} 当前注册的终端数量, N_{mri} 是 GK_{si} 的终端注册容量. P_{ci} 是 GK_{si} 的通话负载因子, $0 \leqslant P_{ci} < 1$, 数值越接近 1, 表明 GK_{si} 的通话负载越重. N_{ci} 是 GK_{si} 当前维护的通话路数, N_{mci} 是 GK_{si} 的呼叫容量. P_{rmin} 是当前区域内所有 GK_s 的最小注册负载因子, P_{cmin} 是当前区域内所有 GK_s 的最小通话负载因子.

为了实现算法, GK_m 需要维护关于区域内所有 GK_s 负载信息的负载表, 其中 GK_{si} 在负载表中的信息表示为: (序号, GK_{si} 地址, GK_{si} 标识, N_{mri}, N_{mci}, N_{ri}, N_{ci}, P_{ri}, P_{ci}). 此外, GK_m 还记录着 P_{rmin} 和 P_{cmin} 分别对应的表项序号 i 和 j 以及负载因子阈值 P_{rt} 和 P_{ct}. 下面的算法描述 GK_m 平衡注册负载:

1) 收到注册请求;
2) 搜索负载表, 找出表项 i;
3) 调用式 (1), 即 $P_{temp} = (N_{ri} + 1) / N_{mri}$;
4) if ($P_{temp} > P_{rt}$)
   向终端返回注册拒绝消息;
   else
   将注册请求转发到 GK_{si} 的地址;
5) $P_{ri} = P_{temp}$;
6) 调用式 (3);
7) 更新 i.

通话负载的平衡算法与上面的算法类似. P_{rt} 和 P_{ct} 由网守管理人员设定, 可以根据系统中终端的数量上下浮动, 以调整网守的最大负载能力. 注册容量 N_r 和呼叫容量 N_c 可以根据对 GK_s 启动时的测试所得, 参考文献[7]提出了一种测试网守注册端点容量和两种测试网守通话容量的方法.

4 结束语

本文描述了一种基于 H.323 协议的分布网守结构, 很好地解决了集中式模型和分层式模型的致命缺点, 为 H.323 系统在大规模多区域网络上的部署提供了较好的解决方案. 为解决 H.323 终端的跨区域呼叫问题, 本文提出了在 GK 层进行信令消息路由的方法. 基于别名的散列设计很好地解决了消息散列表膨胀的问题. 考虑到 GK 的负载平衡问题, 区域内部采用集中式 GK 模型, 主 GK 负责平衡区域内所有从 GK 的负载. 文章最后提出负载因子平衡算法, 提供了较好的平衡 GK 负载方案.

参考文献 (References)

[1] ITU-T Recommendation H.323 V2 (1998), Packet-based multimedia communication systems[S]

[2] Radhika R Roy. Framework for H.323 inter-gatekeeper communications[R]. APC-1385 Q.11-14/16, 1998

[3] 郝又超, 王文东. 网守实现若干问题和策略的研究 (I)[J]. 北京邮电大学学报, 2000, 23(2): 62-66
Hao Youchao, Wang Wendong. Some problems and strategies in the implementation of gatekeeper[J]. Journal of Beijing University of Posts and Telecomm Univ, 2000, 23(2): 62-66(in Chinese)

[4] 张晓, 张德运, 赵东平, 等. 分层式可扩展网守系统的设计与实现[J]. 计算机工程, 2004, 15(30): 97-99
Zhang Xiao, Zhang Deyun, Zhao Dongping, et al. Design and implementation of a hierarchical and scalable gatekeeper system[J]. Computer Engineering, 2004, 15(30): 97-99(in Chinese)

[5] Li Rui, Yu Zhanwu, Wang Hao. Research on multi-zone and hierarchical routing for video gatekeeper based on agent[C]// Proceedings of the 2005 Systems Communications. Montreal: IEEE Computer Society Conference Publishing Services, 2005: 159-164

[6] Radhika R Roy. Distributed gatekeeper architecture for H.323-based multimedia telephony[C]// Proceedings of the 24th Conference on Local Computer Networks. Boston: Lowell Massachusetts, 1999: 73-76

[7] 申艳, 朱志祥, 裴昌幸. 网守容量的测试方法研究[J]. 西安邮电学院学报, 2004, 9(1): 52-55
Shen Yan, Zhu Zhixiang, Pei Changxing. Study on test of gatekeeper capacity[J]. Journal of Xi'an University of Posts and Telecommunications, 2004, 9(1): 52-55(in Chinese)

(责任编辑: 娄 嘉)