互联网名字空间结构及其解析服务研究*

李 丹+, 吴建平, 崔 勇, 徐 恪

(清华大学 计算机科学与技术系,北京 100084)

Research on the Structures and Resolutions of Internet Namespaces

LI Dan+, WU Jian-Ping, CUI Yong, XU Ke

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

+ Corresponding author: Phn: +86-10-62795818 ext 6854, E-mail: lidan@csnet1.cs.tsinghua.edu.cn

Received 2005-03-08; Accepted 2005-06-09

Li D, Wu JP, Cui Y, Xu K. Research on the structures and resolutions of Internet namespaces. *Journal of Software*, 2005,16(8):1445−1455. DOI: 10/1360/jos161445

Abstract: With the rapid growth of Internet, the structures and resolutions of Internet namespaces are facing with new challenges. IP address is overloaded in semantics because it is used to represent both the location and the identity of a device. The namespace resolution system DNS can't satisfy many new application demands with its unique service model, slow update speed, and weak capability in resource description. Many network middleboxes, such as NAT, various agents and firewalls, also destroy the Internet architecture and the Internet namespaces. Based on these, researchers bring forth many solutions to improve the structures and resolutions of Internet namespaces. This paper analyzes the problems in the structures and resolutions of current Internet namespaces, and then makes categories, surveys, and comparisons on the improvement solutions to them. Finally, future research trends on this area are discussed.

Key words: Internet namespace; IP address; domain name; DNS, network middlebox

摘 要: 随着互联网的迅速发展,互联网名字空间结构及其解析服务面临着新的挑战.IP 地址同时用作设备的位置标识和设备的身份标识、语义过载;当前的名字空间解析系统 DNS 存在着服务模式单一、更新速度慢、资源描述能力不够强等缺点,不能满足许多新型应用的要求;而 NAT、各种代理、防火墙等网络中间件的存在也破坏了互联网的体系结构和名字空间.针对这些问题,研究者提出了许多方案来改进互联网的名字空间结构及其解析服务.分析了当前的互联网名字空间结构及其解析服务存在的问题,并对目前的各种互联网名字空间改进方案进行了分类、综述与比较,并展望了进一步的研究方向.

关键词: 互联网名字空间;IP 地址;域名;DNS;网络中间件

* Supported by the National Natural Science Foundation of China under Grant No.90104002 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.2003CB314801 (国家重点基础研究发展规划(973))

作者简介: 李丹(1981－),男,四川阆中人,博士生,主要研究领域为计算机网络体系结构,互联网名字空间,组播;吴建平(1953－),男,教授,博士生导师,主要研究领域为计算机网络体系结构,协议工程学,互联网络;崔勇(1976－),男,博士,助理研究员,主要研究领域为计算机网络体系结构,服务质量控制,路由算法,性能评价;徐恪(1974－),男,博士,副教授,主要研究领域为计算机网络体系结构,路由器体系结构,路由算法与协议,组播与服务质量控制.

中图法分类号: TP316 文献标识码: A

为了对一个分布式系统中的实体进行区分和访问,有必要给其中的每个实体分配一个名字,或者称为标识.一个概念范畴内的实体可以被分配的所有名字的集合,称为名字空间.比如,所有合法的 IPv4 地址的集合,组成了 IPv4 地址的名字空间.

当前互联网采用的是 TCP/IP 五层结构,从下到上依次是物理层、链路层、网络层、传输层和应用层.除物理层以外,每层协议实体都有自己的名字空间,如图 1 所示.链路层实体的名字空间是 MAC 地址,网络层实体的名字空间是 IP 地址(IP address)[1,2],传输层实体的名字空间是(IP 地址,端口号),而应用层实体的名字空间通常是域名(full qualified domain name)[3]及其扩展(如 http 地址、E-mail 地址等).

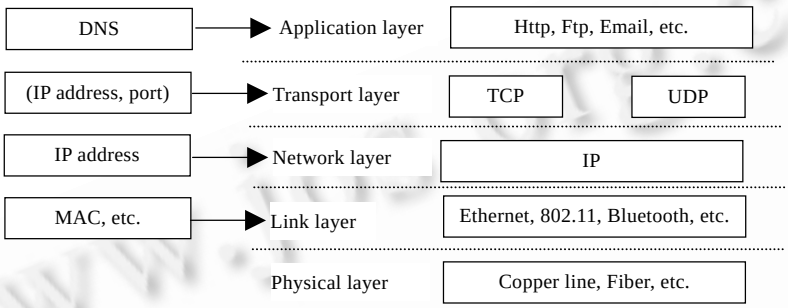

Fig.1 Current Internet namespace

图 1 当前的互联网名字空间

在这些名字空间中,与互联网体系结构和协议关系最密切的是域名和 IP 地址.它们之间是通过域名解析系统(DNS)来进行解析操作的.但互联网的发展给这些名字空间带来了许多新的挑战,它们在语义、功能等多方面的设计上已经不能满足一些新型的互联网应用需求.IP 地址承载了过多的语义,同时作为设备的身份标识和位置标识,使得移动、Multi-homing、IP 地址重分配等问题难以得到很好的解决.现有的 DNS 系统对资源的描述能力不够强,从一个域名到一个 IP 地址的简单解析方式不能适应一些新的应用,而且 DNS 还存在着容易配置出错、记录更新速度慢等一系列问题.NAT[4]、各种代理、防火墙等边缘网络中间件的存在也在一定程度上破坏了互联网的体系结构,使得互联网名字空间结构显得比较混乱.为了解决现有的互联网名字空间在结构及解析服务方面给互联网带来的这些问题,设计更完善、更适合互联网应用需求的名字空间,研究者提出了许多对互联网名字空间结构及其解析服务进行改进的方案.本文综述了这一领域的最新研究进展,并对各种改进方案进行了分类和比较.

本文第 1 节介绍名字空间的分类.第 2 节讨论当前的互联网名字空间结构及其解析服务以及其中存在的问题.第 3~5 节分别综述对互联网名字空间的 3 个方面的改进方案.第 6 节对这些改进方案进行比较,第 7 节总结全文,并展望进一步的研究方向.

1 名字空间的分类

名字空间有 3 个重要的属性[5]:名字的特征,名字的分配以及名字的解析.我们可以根据这些不同的属性对名字空间进行分类.

1.1 名字的特征

名字有多方面的特征,常见的如名字空间的范围、名字长度是否固定以及名字是否有内部结构等.(1) 根据名字空间的范围,可以把名字空间分为全局名字空间和局部名字空间.全局名字空间是指概念范畴内的所有实体都可以直接从一个名字空间中获取唯一的名字;局部名字空间,则指把概念范畴内的实体分为多个区域,不同区域中实体的名字可以复用.(2) 根据名字的长度是否固定,可以把名字空间分为定长名字空间和可变长名字空间.这两种名字空间各有优点.一般来说,计算机对定长名字的处理效率更高,而人们对可变长名字的使用更

加方便.(3) 从名字是否有内部结构来看,名字空间又可以分为扁平名字空间、层次名字空间和组合名字空间.扁平名字空间中的名字是完全无结构的;层次名字空间中名字的组成元素是从逻辑上相继的子空间中获取的,比如域名;而组合名字空间则介于前两者之间,比如基于属性的命名方式,实体的名字是由实体的若干属性组合而成的.

1.2 名字的分配

实体的名字是通过某个权威机构的分配得来的.从名字分配的永久性来说,有些名字是一次分配、永久使用的,而有些名字是动态分配、可以复用的.比如,域名是动态分配的,一个域名可以在被某个主机使用一段时间后分配给另一个完全不同的主机.而身份证号则是永久性的,在人的生命周期内不会被复用.根据名字和实体之间的对应关系,名字的分配还有唯一性和非唯一性之分.如果实体和名字之间的对应关系是一对一的,这种名字分配称为唯一性的.非唯一的名字分配有一对多和多对一两种情况,前者是指一个实体可能被分配多个名字,后者是指出于某种需要对多个实体分配同一个名字.

1.3 名字的解析

名字的解析是指把名字空间中的名字解析为对应的实体,或者把一个名字空间中的名字解析为另一个名字空间中的名字.比如 DNS 就是从域名空间向 IP 地址空间的解析系统.(1) 名字解析可能是全局统一的,也可能是局部相关的.全局统一的名字解析是指解析操作在一个全局唯一的机构上进行.而对于局部相关的名字解析,要查找的对应关系被复制到很多局部机构上,在进行名字解析时直接从这些局部机构上获得解析结果,这种方式也常被称为 cache 机制,比如 DNS 就采用这种机制.显然,全局统一的名字解析结果具有唯一性,但实现效率比较差;局部相关的名字解析实现效率高,但如果对应关系发生变化,解析结果的收敛速度一般较慢,从这些不同的局部机构上获取的结果可能是不一致的,还可能是错误的.(2) 名字解析还有一个比较重要的性质是解析发生的时间,有“早解析”和“晚解析”之分.名字解析往往不是独立发生的,而是伴随着相应的目的操作.比如,DNS 解析的目的操作是发送带 IP 地址的分组;如果把“路由”看作是对 IP 地址的解析,那么其目的操作就是 IP 分组的传递过程.如果解析发生在目的操作之前,则称为“早解析”;否则,称为“晚解析”.因此,源路由可以看成是“早解析”的,而基于路由表的逐跳路由则是“晚解析”的.

2 当前互联网名字空间的结构及其解析服务中存在的问题

2.1 当前互联网名字空间的结构及其解析服务

当前互联网的各层结构中,除了物理层以外,每层协议实体都有自己的名字空间,它们分别是链路层的 MAC 地址、网络层的 IP 地址、传输层的(IP 地址,端口号)和应用层的域名.在这些名字空间中,MAC 地址只要求在子网内唯一,而端口号只要求在主机内部唯一,因此互联网的全局名字空间主要有两个:IP 地址和域名.域名和 IP 地址之间是通过 DNS 进行解析的.

IP 地址不仅作为通信设备的网络接口的身份标识,还作为网络接口在网络拓扑中的位置标识.考虑到路由处理的效率,IP 地址的长度是固定的(IPv4 中是 32 位,IPv6 中是 128 位).为了减少路由表的条目,IP 地址一般都是按照层次结构来组织的,因此,IP 地址是一个有结构的层次名字空间,不同的层次表示 IP 地址在网络拓扑中的位置区域.IP 地址是可以复用的,一段 IP 地址空间可以被用来表示不同的网络拓扑区域,而且在 DHCP[6]等技术中,IP 地址也是动态分配的.一个设备接口可以有多个 IP 地址,因此 IP 地址的分配是非唯一的.如果把路由过程看作是对 IP 地址的解析,那么 IP 地址的解析是全局统一的,因为每个路由器都单独进行全局的路由计算.如前所述,IP 地址的解析既可以是“早解析”(源路由)的,也可以是“晚解析”(逐跳路由)的.

域名表示主机的名字,但在实际使用中常常对其进行扩展以作为应用层的实体标识,如 http 地址、ftp 地址、E-mail 地址等.域名空间是可变长的,而且也是一个层次名字空间,不同的层次反映域名的管理结构.域名是动态分配并可以复用的,一个域名可以分配给某个主机一段时间,然后又分配给另一个完全不同的主机.一个主机可能拥有多个域名,因此域名的分配也是非唯一的.

当前互联网名字空间的解析服务主要是 DNS,负责从域名到 IP 地址的解析.考虑到实现的效率,当前 DNS 的解析操作是局部相关的,即采用 cache 机制.DNS 解析对应的操作一般是发送带 IP 地址的分组,由于分组发送总在 DNS 解析完成之后进行,因此 DNS 解析是"早解析"的.

2.2 存在的问题

在互联网最初的设计中,核心网络采用的是单播和尽力发送的模型,而且认为 IP 地址空间是足够用的,主机是静止的,因此名字空间的设计也相对简单.这种简单性是互联网能够取得今天的巨大成功的重要因素之一.但是随着互联网及其应用的飞速发展,当前的互联网名字空间结构及其解析服务也面临着许多挑战.

IP 地址不仅作为网络层的设备位置标识,而且也作为传输层的设备身份标识,实际上破坏了互联网分层结构中不同层次之间尽量减少耦合的原则,网络层和传输层的紧密耦合不利于每层各自独立地发展.以 IP 地址作为主机的身份标识,也难以实现不同网络区域之间的主机互访,比如,IPv4 网络和 IPv6 网络之间、公有网络和私有网络之间等.IP 地址之所以同时作为设备位置标识和设备身份标识,其重要原因之一是最初互联网的设计并未考虑主机移动的情况.但随着互联网中移动设备越来越多,IP 地址语义过载的缺陷逐渐显露出来.当主机移动时,主机的位置发生变化,IP 地址也相应改变,但主机的身份并未变化.在目前解决移动的主要方案 Mobile IP[7] 中,用主机的家乡地址作为主机的身份标识,通过设置家乡代理来实现路由重定向.但三角路由问题[8,9]、切换延迟问题[8−10]和潜在的安全隐患[9,11]都使 Mobile IP 备受争议.虽然 Mobile IPv6[12]中避免了三角路由,但仍然存在着较大的切换延迟[13].而且 Mobile IP 和 Mobile IPv6 都通过对路由器的修改来解决移动问题,可部署性较差.归根到底,解决移动问题的本质在于把主机的位置标识和主机的身份标识分离开来.除此之外,为了更好地解决 IP 地址动态重分配、Multi-homing[14]等问题,也有必要对 IP 地址的双重功能进行分离.

随着互联网的发展,多种多样的数据和服务成为互联网中的重要对象,而且这些数据和服务的位置及性能可能是经常变化的,这就对当前基于主机域名来对资源进行命名的机制提出了挑战,需要一种命名机制来对网络资源进行与资源位置无关的标识[28].有的应用需要从多种可选的服务中选取一个或多个满足要求的服务,比如,把打印任务发送给一个地理位置最近的负载最轻的打印机,根据网络性能和服务器负载情况选择一个服务器下载文件等.目前的 DNS 解析中把一个域名解析为一个 IP 地址的服务模式显得过于简单,无法有效地支持上述服务.另外,DNS 服务器对域名空间的管理过多地依赖于手动配置,手动配置的错误也会对 DNS 的解析产生很大的负面影响[15].在移动环境中,当数据和服务的位置发生变化时,需要尽可能快地对 DNS 服务器进行记录更新,但目前 DNS 采用的 cache 机制也很难满足这种需求.

私有网络、Ad hoc 网络、传感器网络等多种形式的边缘网络的出现,为互联网带来了 NAT 这样的网络中间件,再加上各种代理、防火墙、DHCP 服务器等网络中间件的存在,在一定程度上破坏了互联网的体系结构和名字空间.由于边缘网络的名字空间与互联网的名字空间并不一致,常常需要在网络中间件进行转换,破坏了互联网的端到端原则[16],可扩展性也比较差.比如,NAT 设备需要在内部网络的私有 IP 地址和互联网的公用 IP 地址之间进行相互转换.网络中间件还往往成为通信中的瓶颈和单一故障点,比如,DHCP 服务器发生故障将使得主机无法获取合法的 IP 地址因而不能进行通信.

针对互联网的发展对名字空间结构及其解析服务提出的挑战,研究者设计了多种对互联网名字空间进行改进的方案.根据解决问题的不同,我们把这些方案大致分为 3 类,第 1 类是对 IP 地址双重功能的分离,第 2 类是对名字空间解析系统的补充和改进,第 3 类是对边缘网络中间件的兼容.下面分别对这 3 类改进方案进行综述.

3 对 IP 地址双重功能的分离

对 IP 地址双重功能的分离,其目的是解决 IP 地址语义过载的问题,把设备的身份标识和位置标识分离开来.这将有助于解决移动、Multi-homing、IP 地址动态重分配以及不同网络区域之间的互访等问题.这一类方案的主要代表有 HIP[17−19],PeerNet[20]和 FARA[22]等.

3.1 HIP

HIP(host identity protocol)[17−19]提出,在域名空间和 IP 地址空间之间加入一层主机标识(host identity)空间.

传输层的连接建立在主机标识上,IP 地址只被用来进行网络层路由,而不再用来标识主机身份.

主机标识是一个抽象概念,在实际中使用的是主机标识符(host identifier).为了与现有的基于 IPv4 地址的协议和应用程序接口兼容,HIP 还定义了局部标识符(local scope identifier),统一长度为 32 位,但只在局部网络范围内使用.因此,在 HIP 中的主机标识层中存在一个抽象的名字空间即主机标识空间,以及 3 个具体的相关名字空间:主机标识符空间、主机标签空间和局部标识符空间.主机标识符空间是全局的、可变长的;主机标签空间是全局的、定长的;而局部标识符空间是局部的、定长的.这 3 个名字空间都是无结构的扁平名字空间.

主机标识空间的名字分配是动态的、非唯一的,因此可以较容易地实现进程迁移和集群服务器.HIP 中的 IP 地址是非唯一分配的,一个主机标识可以对应多个 IP 地址,因此能较好地解决 Multi-homing 和移动的问题.Multi-homing 设备的主机标识对应多个 IP 地址,如果一个 IP 地址不能使用了,或者有更好用的 IP 地址,已经建立的传输层连接可以很容易地转移到其他 IP 地址.结点在移动过程中 IP 地址发生了改变,而主机标识并没有变,因此传输层的连接可以不中断,但是移动结点应该通知自己 IP 地址的改变.由于传输层连接是与主机标识绑定的,IP 地址只是被用来进行路由,因此 HIP 还可以实现不同网络区域之间的互访,如公有网络与私有网络之间、IPv4 网络与 IPv6 网络之间等.

但 HIP 中主机标识层的名字空间定义太复杂,在实际使用中要维护很多对应关系,增大了管理开销和出错的概率.而且这些名字空间都是无结构的,查找效率低.HIP 中也没有提供一个从主机标识空间到 IP 地址空间的全局解析机制.

3.2 PeerNet

PeerNet[20]也提出把节点的身份标识和位置标识进行分离,避免 IP 地址同时拥有双重功能.但 PeerNet 是一个全新的网络层协议,它的目的是取代 IP 协议.PeeerNet 网络中的每个节点都有一个节点标识和一个地址标识.节点标识可靠地代表节点的身份,不随节点的移动而改变.地址标识则严格地反映节点在网络中的当前位置,当节点移动时,节点的地址标识随之动态改变.由于地址标识是完全层次化分配的,PeerNet 中就可以运用 P2P 网络的路由思想来组织路由[21].

因此,PeerNet 中实际上引入了两个新的全局名字空间,以取代现在的 IP 地址空间.节点标识空间是全局的、非唯一分配的,但对节点标识空间的其他性质,PeerNet 中并没有进行明确定义.地址标识空间是全局的、固定长度的、层次化结构的;而且是唯一分配的,一个节点虽然可能负责管理多个地址,但只能使用反映当前网络拓扑位置的唯一地址标识;如果把网络拓扑位置认为是网络层地址所表示的实体,则可以认为地址标识的分配是一次性的.

PeerNet 中节点标识空间到地址标识空间的解析是全局统一的、“早解析”的.每个节点的节点标识和地址标识之间的对应关系由某个其他节点来维护,对于任何一个节点来说,其对应关系的维护节点在任何时刻都是唯一的.在进行通信之前,源节点只需知道目的节点的节点标识,再利用解析机制查找到目的节点的地址标识,然后发送分组.由于地址标识仅用于网络层的路由,因此 PeeNet 中可以较好地支持移动.当移动节点获取新地址后,应该及时地进行节点标识和地址标识对应关系的更新.PeerNet 还可以支持组播和任意播.

PeerNet 中也存在许多问题.首先,很难保证 P2P 路由表中的紧邻项也位于实际网络的紧邻位置,这对路由表的维护和路由效率都有影响;其次,由于每个节点只能使用反映当前网络拓扑的唯一地址,PeerNet 不能支持 Multi-homing.

3.3 FARA

FARA(forwarding directive, association, and rendezvous architecture)[22]基于把节点标识与网络层地址分离的思想,定义了一种具有广泛性和灵活性的抽象模型.在这种抽象模型的基础上,可以根据具体需求设计更加具体的体系结构,比如支持移动的体系结构(M-FARA)等.

在 FARA 中,主机与主机之间的通信概念不再存在,取而代之的是实体与实体之间通过底层分组交换结构进行的连接(association).FARA 中用不同的 Association ID(AID)来标识一个实体的多个不同连接,由于 AID 只在每个实体上是唯一的,所以分组携带一个(源 AID,目的 AID)对.显然 AID 是一个局部名字空间,类似于现在传

输层的端口号.FARA 中的实体依赖于底层的分组交换结构来进行通信,包括操作系统和网络.FARA 中用转发指示(forwarding directive)作为分组交换结构的地址标识.在进行通信时,实体需要通过另外的机制找到目的实体的转发指示.

FARA 把底层网络的转发机制和实体之间的端到端通信功能完全分离开来,允许底层转发结构和端到端应用各自独立地发展.可以自由地改变一个会话的转发指示,既能在不同网络区域之间进行互访,也可以解决移动问题.FARA 也没有对如何查找实体的转发指示进行规定,这种机制可能是全局统一的,也可能是局部相关的.

FARA 实际上是一种重新设计互联网体系结构的思想,其模型是非常抽象的.但 FARA 中的通信实体并没有一个全局标识,需要带外的机制来寻找通信实体的位置,这在实际应用中并不容易做到.而且对于如何进行移动切换、保证通信安全等许多问题,FARA 中并没有具体的实现机制.

4 对名字空间解析系统的补充和改进

当前互联网名字空间的解析系统主要是 DNS.目前的 DNS 系统存在着记录更新速度慢、服务模式单一、资源描述能力不够强、配置易出错等缺点,因此出现了许多对 DNS 进行补充和改进的方案,包括 URN,INS 和 CoDoNS 等.

4.1 URN

URN(uniform resource names)[23−26]的目的是对互联网的所有资源进行统一的命名,称为统一资源名.传统的域名实际上是与资源位置即资源所在的主机相关的,而统一资源名是与资源位置无关的,在通信过程中再把统一资源名解析为域名.因此可以把统一资源名看作是比域名更高一级的名字空间,把 URN 看作是对 DNS 的补充.

URN 中的统一资源名空间是全局的、可变长的.统一资源名由 3 部分组成:“urn:”〈NID〉“:”〈NSS〉,其中 NID 表示子名字空间标识,NSS 表示子名字空间中的特定标识符.URN 的分配是永久性的、唯一的,每个资源都拥有一个永远使用的唯一标识.为更好地实现负载均衡并抵御 DoS 攻击,URN 的解析同 DNS 一样,也是局部相关的.

DNS 对域名的解析是层次化进行的,在解析过程中各个管理域解析的名字越来越本地化;但在 URN 中,统一资源名可能一次性地全部解析.因此,与 DNS 相比,URN 的名字解析在复杂性、解析失败的可能性等方面都大大增加了.而在 URN 的第 1 个公开发布的实现[27]中,仍然采用了 DNS 的框架,因此仍然具有 DNS 系统本身的一些局限性.

4.2 INS

INS(intentional naming system)[28]是一个解决移动环境中的资源发现和服务定位问题的命名系统,是对 DNS 的补充.INS 的主要特征有 4 个:一是资源描述名的表达能力强,能够描述大量的服务和设备;二是反应迅速,当设备或服务的位置、性能发生变化时能够快速作出反应;三是鲁棒性好,当某个服务节点或某个名字解析器发生故障时,名字系统能进行及时的调整以保证正常工作;四是易于配置,无须太多的手工参与.

INS 的资源描述名是一个全局的、可变长的名字空间.为了具有较强的表达能力,INS 用基于属性和数值的组合方法对服务和设备进行命名,因此 INS 的资源描述名空间是组合结构的.当网络中的服务或者服务所在的设备发生移动或者服务的性能发生变化而导致最佳服务所在的位置发生变化时,INS 解析器的解析结果应该立即作出反应.为了实现这一点,INS 支持“晚解析”的名字解析方式.解析器不仅负责把资源描述名解析为资源所在的位置,而且还在解析器之间进行基于资源描述名的路由.这样能以最快的速度反映设备或服务的变化情况.为了实现鲁棒性,INS 的名字解析是局部相关的.并且 INS 在服务节点和解析器之间以及在各个解析器之间进行周期性的服务通告和基于软状态的消息交换[29],以此提高解析性能,并避免出现单一故障点.为了易于配置,INS 的所有解析器组成一个自组织的覆盖网络.

INS 是 DNS 的补充,而不是为了取代 DNS.DNS 中的域名是基于静态的层次结构的,而 INS 中的资源描述名是基于属性和数值的组合的;DNS 中的名字解析器只有解析功能,而 INS 中的名字解析器还引入了名字路由功能;INS 中的服务器节点对其提供的资源描述名还要进行动态的通告,并且名字解析器之间组成一个自组织

的覆盖网络.但 INS 的管理开销要远远大于 DNS,而且 INS 与 DNS 之间如何配合使用还有待研究.

4.3 CoDoNS

CoDoNS(cooperative domain name system)[30]的设计目的是逐步取代传统的 DNS 系统.其主要特点有 3 个:一是有较高的解析性能,解析时间比传统 DNS 明显减少;二是能较好地抵抗 DoS 攻击;三是能够进行快速更新.

在 CoDoNS 中,每个管理域都有一个或几个 CoDoNS 服务器,所有管理域的 CoDoNS 服务器共同组织成一个有结构的 P2P 自组织网络.而每个管理域内仍然采用传统 DNS 机制.CoDoNS 服务器上的记录可以从各个管理域内的传统 DNS 服务器上获取,也可以进行手动配置.利用有结构的 P2P 网络的性质,CoDoNS 服务器网络对域名空间的管理更加方便,而且可以有效地实现错误恢复.

在 CoDoNS 服务器组成的 P2P 网络中,资源(记录)和服务器都被哈希映射到一个定长的名字空间,并采用分布式哈希表[31]的方式进行存储和路由.CoDoNS 服务器网络还通过一种 Beehive[32]的机制来提高查询速度.Beehive 采用预先缓存的方法,把资源在多个 CoDoNS 服务器上进行复制并缓存,可以在付出有限的存储代价的情况下大大减少域名查询的路由跳数,因此明显减少了 DNS 的解析时间,提高了解析性能.通过在多个服务器上进行缓存,CoDoNS 可以比传统 DNS 更有效地抵抗 DoS 攻击,减小恶意用户对 DNS 系统的威胁,还能有效地实现负载均衡,并且避免出现单一故障点.CoDoNS 把有组织的 P2P 网络和预先缓存机制相结合,取代了传统 DNS 中基于层次的 cache 机制,也就可以实现比传统 DNS 更快速的记录更新.

CoDoNS 仍然保留了传统 DNS 中的层次域名空间,但把域名空间管理和域名解析分离开来.CoDoNS 用对扁平的 P2P 网络标识空间的管理取代了传统 DNS 中对层次结构的域名空间的管理,克服了传统 DNS 的一些缺点.但如果要把域名的解析时间减少到很短的话,CoDoNS 服务器也要付出可观的存储开销.

4.4 其他方案

对 DNS 的补充和改进是目前互联网研究的热点之一,除了上面介绍的方案,还有一些其他方案,这里只作简要的介绍.Cohen 和 Kaplan[33]提出对 DNS 记录进行预先缓存.对 cache 中失效的记录,在收到 DNS 查询请求之前就从更高级的服务器上获取更新信息,这样可以提高查询的效率.CoDNS[34]在本地名字服务器失效时把查询请求转移到其他的有效名字服务器,从而减小本地名字服务器失效引起的查询延迟.DDNS[35]和 Overlook[36]都是基于 P2P 结构的名字解析服务,DDNS 基于 Chord[37],而 Overlook 基于 Pastry[38].它们都利用 P2P 网络的性质增强了名字解析服务的容错能力,实现负载均衡,但却增加了查询延迟.secure DNS update[39,40]针对当前 DNS 系统记录更新速度慢的问题,设计了 DNS 的快速动态更新机制.

5 对边缘网络中间件的兼容

NAT、各种代理、防火墙、DHCP 服务器等边缘网络中间件的存在一定程度上破坏了互联网的体系结构和名字空间.边缘网络的名字空间和互联网的名字空间常常不一致,而且网络中间件往往是通信的单一故障点.为了维护互联网的端到端原则,研究者对互联网名字空间进行改进以从网络体系结构上兼容各种网络中间件.这些方案往往结合对互联网名字空间的其他改进一起来实现,如 IPNL,UIP 以及 LNAI 等.

5.1 IPNL

IPNL(for IP next layer)[41]是一种基于 NAT 扩展的互联网体系结构.IPNL 的设计目的主要有两个,一是实现公有网络和私有网络之间的互访,以支持 P2P 等应用;二是不再在 NAT 网关上维护每流的状态,以消除 NAT 可扩展性差的缺点.其解决办法是在互联网协议族的 IP 层之上、传输层之下再加一个 IPNL 层.IPNL 层的协议头部结构包括一个必需的局部地址头、一个可选的全局地址头和一个可选的域名头.局部地址头填充 NAT 内部主机的私有 IP 地址,全局地址头填充 NAT 网关的公有 IP 地址,域名头填充主机域名,每个头部又包括源和目的两部分.可见,IPNL 层实际上使用了 3 个名字空间,即公有 IP 地址空间、私有 IP 地址空间和域名空间.公有 IP 地址空间和主机域名空间的特点前面已经作了介绍,而私有 IP 地址空间显然是一个局部名字空间.

通过增加 IPNL 层,私有网络与互联网之间以及私有网络之间可以实现相互访问,可以支持 P2P 应用以及服

务器在内部网络中的应用.而且,NAT 网关不再需要地址池来维护私有地址和公有地址之间的对应关系,解决了 NAT 可扩展性差的问题.与 IPv6 相比,IPNL 有利于解决网络接入地址重分配及 Multi-homing 等问题,但 IPNL 显然大大增加了路由的复杂性.

5.2 UIP

UIP(unmanaged internet protocol)[42]的设计目的与 IPNL 有相似之处,都是为了更好地管理互联网的边缘网络.而且 UIP 与 IPNL 一样,在互联网核心网络中仍然使用 IP 地址来进行路由.但与 IPNL 不同的是,UIP 不再用 IP 地址作为边缘网络节点的身份标识,而引入了一个新的节点标识来标识节点的身份,并在边缘网络进行基于节点标识的路由.

UIP 中的节点标识是一个全局名字空间,与 HIP 中的主机标识符很类似.UIP 中的节点标识被用来在边缘网络进行基于分布式哈希表的路由,相当于边缘网络设备组成了一个覆盖网络.

UIP 解决了边缘网络难以管理的问题,在 DHCP 服务器失效、边缘设备发生移动等多种情况下,都能保证边缘网络通信的顺利进行.UIP 还可以应用到穿越不同网络区域的通信中.但 UIP 还只停留在初步设想阶段,许多技术细节都还没有解决,包括节点标识的分配和解析机制、UIP 的协议细节等.

5.3 LNAI

LNAI(a layered naming architecture for the internet)[43]对互联网名字空间的改进实际上包括了对 IP 地址双重功能的分离、对 DNS 的改进和对边缘网络中间件的兼容 3 个方面.LNAI 提出了互联网的 4 层名字空间结构,分别是用户级描述符(如搜索关键字、E-mail 地址等)、服务描述符(SID)、主机描述符(EID)和 IP 地址.因此也需要 3 层解析器,从用户级描述符到服务描述符的解析,从服务描述符到主机描述符的解析,以及从主机描述符到 IP 地址的解析.LNAI 的主要特点是:1) 通过对服务和数据进行永久性的命名(SID),使它们成为互联网上最重要的对象;2) 通过引入 EID,不再用 IP 地址作为主机的标识,从而可以无缝地支持移动和 Multi-homing;3) 通过各个层次的重定向功能,把边缘网络中间件整合到互联网的体系结构中来.

LNAI 是根据 4 个设计原则来进行设计的:1) 每层的名字只能表示该层的实体,而不能反映不直接相关的底层细节.目前的互联网体系结构就破坏了这个原则.当应用程序请求一个服务时,关心的只是服务本身,而并不关心服务所在的主机.但现在的 DNS 系统把应用程序的请求与 IP 地址直接绑定,相当于对这个原则进行了双重破坏,不但把服务和服务所在的主机绑定,而且把主机和主机的 IP 地址绑定.根据这一原则,要在现在的互联网中加入两层名字空间,即 SID 和 EID,而传输层的连接建立在 EID 上.2) 名字不应该对它所表示的实体加任何限制.目前互联网中的 IP 地址和域名都破坏了这个原则.IP 地址反映了拓扑结构,而域名反映了管理结构.根据这个原则,LNAI 中的 SID 和 EID 都采用扁平的无结构的名字空间.3) 每层实体可以把自己的名字的解析结果重定向为它所选择的代理的位置.4) 每层的通信接收者都可以从单一接收者推广为接收者序列,类似于网络层的源路由.这样,每层的通信实体都可以指定该层的通信数据所经过的路径.

LNAI 的部署并不需要改变互联网的底层结构,但需要对现有的主机软件作很大的改变,包括协议和应用程序,而且解析这些扁平的名字空间需要新的解析器.因此,LNAI 的部署代价还是比较大的.在引入了多个层次的名字空间后,安全问题如 DoS 攻击也成为了 LNAI 中的一个比较大的隐患.

6 各种方案的比较

我们对本文综述的各种改进互联网名字空间结构及其解析服务的方案与当前名字空间一起进行比较,比较方面包括解决的问题、部署代价、引入的新名字空间以及新名字空间的特征等方面,见表 1.

从表 1 可以看出,每种方案对互联网名字空间的改进角度各不相同,但都引入了一个或几个新的名字空间.总体来说,PeerNet 和 FARA 对互联网名字空间的改动最大,改变了核心网络的 IP 地址结构,其部署代价是非常大的;HIP 和 LNAI 虽然只需在端系统上实现,但由于引入了新的传输层标识,需要对所有的现有主机软件进行改动,部署代价也比较大;URN 和 INS 都需要在全网一次性成功部署,部署代价与其他方案比起来相对适中;而 IPNL 和 UIP 的部署只需改动部分边缘网络的设备,CoDoNS 可以逐步地进行部署,它们的部署代价是这些方案

中相对较小的.

Table 1 The comparison of improvement solutions to the structures and resolutions of Internet namespace

表 1 互联网名字空间结构及其解析服务改进方案比较

Solution	Class 1	Class 2	Class 3	Deployment cost	Namespaces	Characteristic of names	Assignment of names	Resolution of names
Current namespaces	–	–	–	–	IP address	Global, fixed-size, hierarchical	Dynamical, non-unique	Global, both early and late resolution
					Domain name	Global, variable-size, hierarchical	Dynamical, non-unique	Local, early resolution
HIP	√	–	–	High	Host identifier	Global, variable-size, flat	Dynamical, non-unique	Early resolution
					Host tag	Global, fixed-size, flat	Dynamical, non-unique	Early resolution
					Local identifier	Local, fixed-size, flat	Dynamical, non-unique	Early resolution
PeerNet	√	–	–	High	Node identifier	Global	Non-unique	Global, early resolution
					Node address	Global, fixed-size, hierarchical	Permanent, unique	Global, early resolution
FARA	√	–	–	High	Association ID	Local	–	–
					Forwarding directive	Local	–	–
URN	–	√	–	Medium	Uniform resource name	Global, variable-size, composite	Permanent, unique	Local, early resolution
INS	–	√	–	Medium	Intentional name	Global, variable-size, composite	Dynamical, non-unique	Local, both early and late resolution
CoDoNS	–	√	–	Low	CoDoNS server identifier	Global, fixed-size, flat	Dynamical, non-unique	Local, early resolution
IPNL	–	–	√	Low	Public IP address	Global, fixed-size, hierarchical	Dynamical, non-unique	Global, both early and late resolution
					Private IP address	Local, fixed-size, hierarchical	Dynamical, non-unique	Global, both early and late resolution
					Domain name	Global, variable-size, hierarchical	Dynamical, non-unique	Local, early resolution
UIP	√	–	√	Low	Node identifer	Global, fixed-size, flat	–	–
LNAI	√	√	√	High	User-level descriptor	–	–	–
					Service ID	Global	Dynamical, non-unique	Early resolution
					Endpoint ID	Global	Dynamical, non-unique	Early resolution

Class 1: 对 IP 地址双重功能的分离;Class 2: 对名字空间解析系统的改进;Class 3: 对边缘网络中间件的兼容.

7 总结和展望

IP 地址和域名是目前互联网的两个主要的全局名字空间,但互联网的发展对它们提出了挑战,主要体现在 IP 地址语义过载、名字空间解析系统 DNS 存在许多缺陷以及边缘网络中间件对互联网体系结构和名字空间的破坏这 3 个方面.在研究者提出的许多改进互联网名字空间的方案中,一般都着眼于解决当前互联网名字空间的一方面或几方面的问题,通过引入新的名字空间来弥补现有名字空间的不足.HIP,PeerNet,FARA 等方案主

要解决 IP 地址语义过载的问题,对 IP 地址的双重功能进行分离.把设备的身份标识和位置标识分离开来,更有利于从体系结构上解决移动、Multi-homing、IP 地址重分配以及不同网络区域之间的通信等问题;而且也把传输层的实体标识和网络层的实体标识分离,这样更符合互联网的分层结构原则,使得传输层和网络层可以各自独立地发展.URN,INS,CoDoNS 等方案主要针对现有名字空间解析系统 DNS 记录更新速度慢、服务模式单一、资源描述能力不够强等缺点,从某些方面对 DNS 进行补充和改进.这些改进方案或者与 DNS 共存以作为 DNS 的补充,或者作为对 DNS 的替代方案以逐步取代 DNS.IPNL,UIP,LNAI 等方案的主要设计目的(或主要设计目的之一)则是兼容当前各种边缘网络中间件,通过改进名字空间的设计把这些网络中间件整合到统一的互联网体系结构中来.

但对互联网名字空间的改进是一项庞大的工程,需要考虑到诸多方面的问题.目前的这些改进方案都还存在着各自的不足,许多问题有待进一步的研究.1) 对于 IP 地址双重功能的分离,在引入设备的身份标识空间后,如何设计新名字空间与 IP 地址之间的解析机制以及如何与现有 DNS 协同工作,在目前的这些方案中还缺乏完整的设计,需要进一步的考虑.2) 在对名字空间解析系统的补充和改进方案中,一个很重要的问题是补充的名字空间如何与当前 DNS 进行无缝的配合,这在已有方案中并没有得到明确的体现.另外,对 DNS 安全性的研究还有待深入.3) 现有的兼容边缘网络中间件的方案大多采用了基于名字路由的思想,增加了路由的复杂性,降低了路由的性能.如何在尽量减小路由复杂性的前提下兼容边缘网络中间件,也是下一步的研究方向.

References:

[1] Internet Protocol. RFC791, 1981.

[2] Deering S, Hinden R. Internet Protocol, Version 6 (IPv6) Specification. RFC2460, 1998.

[3] Mockapetris PV. Development of the domain name system. In: Proc. of the ACM SIGCOMM. 1988. 123−133.

[4] Egevang K, Francis P. Traditional IP network address translator (Traditional NAT). RFC3022, 2001.

[5] Sollins K. Recursively Invoking Linnaeus: A Taxonomy of Distributed Naming Systems. 2003.

[6] Droms R. Dynamic host configuration protocol. RFC2131, 1997.

[7] Perkins CE. IP mobility support for IPv4. RFC3220, 2002.

[8] Malki KE. Low latency handoffs in mobile IPv4. Internet draft, draft-ietf-mobileip-lowlatency-handoffs-v4-09, 2004.

[9] Perkins CE. Mobile networking in the Internet. ACM Mobile Networks and Applications (MONET), Special Issue on Mobile Networking in the Internet, 1998,3(4):319−334.

[10] Koodli R. Fast handovers for mobile IPv6. Internet draft, draft-ietf-mobileip-fast-mipv6-08, 2003.

[11] Perkins CE, Calhoun PR. AAA registration keys for mobile IP. Internet draft, draft-ietf-mobileip-aaa-key-13, 2003.

[12] Johnson D. Mobility support in IPv6. RFC3775, 2004

[13] Koodli R. Fast handovers for mobile IPv6. Internet draft, draft-ietf-mipshop-fast-mipv6-03, 2004.

[14] Bates T. Scalable support for multi-homed multi-provider connectivity. RFC2260, 1998.

[15] Pappas V, Xu Z. Impact of configuration errors on DNS robustness. In: Proc. of the ACM SIGCOMM. 2004. 319−330.

[16] Saltzer J, Reed D, Clark D. End-to-End arguments in system design. ACM Trans. on Computer Systems, 1984,2(4):277−288.

[17] Moskowitz R. Host identity protocol. Internet draft, draft-moskowitz-hip-09, 2004.

[18] Moskowitz R. Host identity protocol architecture. draft-moskowitz-hip-arch-06, 2004.

[19] Nikander P. Integrating security, mobility, and multi-homing in a HIP way. In: Proc. of the Network and Distributed Systems Security Symp. (NDSS 2003). 2003. 87−99.

[20] Eriksson J. PeerNet: Pushing peer-to-peer down the stack. In: Proc. of the Int'l Workshop on Peer-To-Peer Systems 2003 (IPTPS 2003). 2003. 268−277.

[21] Hong X, Xu K, Gerla M. Scalable routing protocols for mobile ad hoc networks. IEEE Network, 2002,6(4):11−21.

[22] Clark D. FARA: Reorganizing the addressing architecture. In: Proc. of the ACM SIGCOMM. 2003. 313−321.

[23] International DOI Foundation. http://www.doi.org/

[24] Sollins K. Architectural principles of uniform resource name resolution. RFC2276, 1998.

[25] Sollins K, Masinter L. Functional requirements for uniform resource names. RFC1737, 1994.

[26] Moats. R. URN syntax. RFC2141, 1997.

[27] Daniel R. Resolution of uniform resource identifiers using the domain name system. RFC2168, 1997.

[28] Winoto WA. The design and implementation of an intentional naming system. Operating Systems Review, 1999,34(5):186−201.

[29] Clark D. The design philosophy of the DARPA internet protocols. In: Proc. of the ACM SIGCOMM. 1988. 106−114.

[30] Ramasubramanian V, Sirer EG. The design and implementation of a next generation name service for the Internet. In: Proc. of the ACM SIGCOMM. 2004. 331−342.

[31] Ratnasamy S, Shenker S, Stoica I. Routing algorithms for DHTs: Some open questions. In: Proc. of the Int'l Workshop on Peer-To-Peer Systems 2002 (IPTPS 2002). 2002. 45−52.

[32] Ramasubramanian V, Sirer EG. Beehive: Exploiting power law query distributions for *O*(1) lookup performance in peer to peer overlays. In: Proc. of the USENIX Symp. on Networked Systems Design and Implementation 2004 (NSDI 2004). 2004. 331−342.

[33] Cohen E, Kaplan H. Proactive caching of DNS records: Addressing a performance bottleneck. In: Proc. of the 2001 Symp. on Applications and the Internet 2001 (SAINT 2001). 2001. 85−94.

[34] Park K, Wang Z, Pai V. CoDNS: Masking DNS delays via cooperative lookups. Princeton University Computer Science Technical Report TR-690-04, 2004.

[35] Cox R, Muthitacharoen A, Morris R. Serving DNS using a peer-to-peer lookup service. In: Proc. of the Int'l Workshop on Peer-To-Peer Systems 2002 (IPTPS 2002). 2002. 155−165.

[36] Theimer M, Jones M. Overlook: Scalable name service on an overlay network. In: Proc. of the Int'l Conf. on Distributed Computing Systems 2002 (ICDCS 2002). 2002. 52−61.

[37] Stoica I, Morris R, Karger D, Kaashoek F, Balakrishnan H. Chord: A scalable peer-to-peer lookup service or internet applications. In: Proc. of the ACM SIGCOMM 2001. 2001. 149−160.

[38] Rowstron A, Druschel P. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. In: Proc. of the Int'l Conf. on Distributed Systems Platforms 2001 (Middleware 2001). 2001. 329−350.

[39] Vixie P, Thomson S, Rekhter Y. Dynamic updates in the domain name system (DNS UPDATE). RFC2136, 1997.

[40] Wellington B. Secure domain name system (DNS) dynamic update. RFC3007, 2000.

[41] Francis P. IPNL: A NAT-extended internet architecture. In: Proc. of the ACM SIGCOMM 2001. 2001. 69−80.

[42] Ford B. Unmanaged internet protocol: Taming the edge network management crisis. In: Proc. of the ACM Hotnets Workshop 2003. 2003. 93−98.

[43] Balakrishnan H. A layered naming architecture for the Internet. In: Proc. of the ACM SIGCOMM 2004. 2004. 343−352.