IPv6 中 Anycast 的一种加权通信模型

王晓喃[1,2]　钱焕延[1]

(南京理工大学　南京 210094)[1]　(常熟理工学院　江苏常熟 215500)[2]

摘　要　本文提出了一种新的 Anycast 加权通信模型，此模型在某种程度上解决了应用层和 IP 层实现 Anycast 技术所存在的问题，如扩展性以及客户与服务器通信失败等等，本文对该 Anycast 的加权通信模型进行了深入分析和讨论，并验证了该模型的可行性及其有效性。

关键词　IPv6，Anycast，Unicast，路由器

Design and Implementation of Anycast Communication Model in Ipv6

WANG Xiao-Nan[1,2] QIAN Huan-Yan[1]

(Nanjing University of Science & Technology, Nanjing 210094)[1]

(Changshu University of Science & Technology, Jiangsu, Changshu 210094)[2]

Abstract　A new kind of anycast weighted communication model is created in this paper and it solves some existing problems, such as scalability and communication errors between clients and servers , which are caused by performing anycast services on application layer or IP layer. The Anycast weighted communication model is deeply analyzed and discussed, and its feasibility and validity are proved in this paper.

Keywords　Ipv6, Anycast, Unicast, Router

1　前言

Anycast 是 IPv6 所提供的一种特殊网络服务，它与 Unicast 一样都是 IP 的一种通信模式。Unicast 使源结点可以向一个单个目的结点发送数据报，该目的结点由一个 Unicast 地址标识；而 Anycast 也是使源结点向一个单个目的结点发送数据报，但是，这个目的结点来自于一个目的结点集合，该目的结点集由一个 Anycast 地址标识，数据报会被路由到该目的结点集中离源结点最近的(根据路由协议的距离量度)一个结点。由此可见，Unicast 只是 Anycast 的一个特例，如图 1 所示。

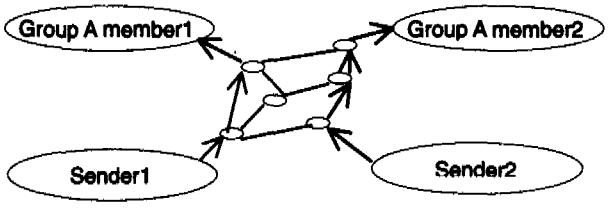

图 1　Anycast 服务

目前，Anycast 技术可以在 IP 层实现，也可以在应用层实现，但是无论在哪层实现都存在着一些问题。在应用层实现 Anycast 技术存在着复杂性和规模性问题，因为在应用层上实现 Anycast 服务需要收集服务器主机是否在线和该主机提供何种服务类型以及每个潜在客户和各个不同服务器间以用户指定的标准度量的距离信息。为了收集这两种信息，Anycast 应用程序不得不频繁地探测服务器的状态，或者服务器频繁地向 Anycast 应用程序报告本机状态。当网络中的业务十分繁忙并且服务器数量很多时，网络可能不堪重负而导致整个网络瘫痪。在 IP 层实现 Anycast 技术的问题是由于当前网络中的路由表变化频繁，可能造成客户和服务器之间的通信障碍。例如，主机 A 在 t_1 时刻和 Anycast 地址为 $Addr_1$ 的主机 $Host_1$ 通信。在 $t_1+\Delta t$ 时刻由于路由的变化，数据包可能被传给拥有此 Anycast 地址的主机 $Host_2$。由于主机 $Host_2$ 没有和主机 A 通信的上下文，所以通信出现问题。此外，IPv6 中 Anycast 地址是从 Unicast 地址空间分配而来的，在语法上 Anycast 地址与 Unicast 地址没有区别，所以这就给 Anycast 的路由带来问题。如果一个 Anycast 地址表示的共享某个特性的结点组分散在互联网的各个地方，那么只能用 Unicast 路由协议路由 Anycast，这样每个全球 Anycast 地址必须作为独立的路由表项处理。这种要求使得路由表会随全球 Anycast 组数成比例增长，从而导致路由表会迅速地膨胀。为了解决这个问题，IPv6 将每个 Anycast 组成员限制在共享一个地址前缀的特殊拓扑区内，在这个拓扑区域中，Anycast 地址在单播路由系统中是独立的表项，在该拓扑区域外，Anycast 地址被汇聚到其所在区域的地址前缀的路由项中传播，但是这种把 Anycast 组限制在一个预定义的区域内的做法，大大限制了 Anycast 组成员在整个网络中的广泛分布，进而影响了 Anycast 的服务质量。在 IP 层实现 Anycast 服务的另外一个缺点是：Anycast 成员的距离度量只能通过 Hop 的次数来衡量，但是有些时候，这些距离需要用其它的度量方式，例如 CPU 负载或者服务器负载等形式，进而影响了 Anycast 的服务质量。

针对以上出现的问题，本文提出了一种新的加权通信机制，该通信机制大大地提高了 Anycast 的服务能力。

钱焕延　教授，博士导师，研究方向为网络安全与应用。王晓喃　博士研究生，研究方向为网络安全与应用。

2 Anycast 通信模型

Anycast 是一种通信模型，一个 Anycast 地址被分配给提供同一种服务的一组节点，发送到这个 Anycast 地址的数据包可以被路由到距离最短的 Anycast 组成员中去。这里的最短距离通常由所使用的路由协议来确定，一般可以包括 Hop 数、服务器负载、到服务器的往返时间(RTT)以及当前可用的带宽等等。但是，实际上用户并不关心这些参数，他们真正感兴趣的是从发送服务请求到接收到服务应答之间的这段时间间隔，即总体应答时间(TRT，Total Response Time)，这段时间越短，客户认为服务质量越好。所以，本模型所采用的距离度量策略是 TRT，因为这个参数不仅反映了服务器负载的繁忙状态，也反映了网络本身以及所建立的连接的某些属性(例如，带宽以及用户到服务器的 Hop 数)，所以，TRT 是一个综合性参数。本模型的体系结构如图 2 所示。

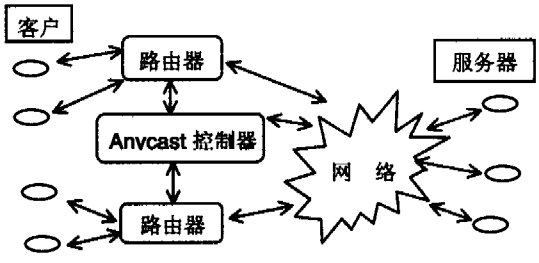

图 2 Anycast 模型体系结构

从图 2 可以看出，本模型在相邻网络之间增设了一个 Anycast 控制器，并与这些相邻网络的路由器直接相连。Anycast 控制器的主要功能是维护整个网络内的 Anycast 组成员的相关数据。客户请求 Anycast 服务的过程可以描述如下：(1)客户提出域名解析申请，并根据域名的后缀(即一级域名设置为 Any，例如：www. njust. edu. any)来判断该服务是否为 Anycast 服务类型；(2)DNS 服务器接收该请求并对其进行解析，同样也根据其一级域名是否为 Any 来判断该服务是否为 Anycast 服务类型，如果是，就返回 Anycast 地址，否则，返回服务器的 Unicast 地址；(3)客户接收到返回的地址之后，根据(1)的判断，如果是 Anycast 服务类型，就将该 Anycast 地址以及自身的 Unicast 地址一起发送到 Anycast 控制器进行 Anycast 地址——Unicast 地址的解析，否则就将其发送到路由器，进行正常路由；(4)Anycast 控制器接收到 Anycast 地址解析请求之后，搜索本机的数据库，如果查找到对应的 Anycast 表项，那么首先根据客户端的 Unicast 地址计算出它与每个 Anycast 组成员的权值，然后再把当前总体响应时间与权值乘积最小的 Anycast 组成员所对应的 Unicast 地址返回给客户端，如果没有查找到该表项，则向其相邻的 Anycast 控制器发送查询消息，然后用查询结果更新本机数据库，并选取最佳的 Anycast 组成员，将其 Unicast 地址返回给客户端；(5)客户端接收到该 Unicast 地址之后，直接与该服务器建立连接；(6)本次服务结束之后，客户端把本次服务的 TRT，RTT 以及传输的总字节数封装在一个数据包中，将其发送给 Anycast 控制器；(7)Anycast 控制器接收到该数据包之后，利用其参数更新自己的数据库。

在本模型中，路由器的功能不需要修改，但是需要增加 DNS 服务器以及客户端的某些功能。下面具体介绍本模型的实现过程。

3 Anycast 通信模型实现

本通信模型不仅解决了 IP 层实现 Anycast 技术所存在的可扩展问题，同时也解决了应用层实现 Anycast 技术的资源高消耗问题。本模型在与现有网络应用程序和协议兼容的基础上只对 DNS、服务器以及客户软件稍作修改，并且在相邻网络之间增加了 Anycast 控制器，但是，路由器的功能保持不变，不做任何修改。

3.1 DNS 服务器

在本模型中，DNS 服务器需要增加解析 Anycast 域名的功能。由于 DNS 中的域名地址是分层结构的，所以，为了区分 Anycast 域名与其他域名，将 Anycast 域名的一级域名设置为 Any，例如：www. njust. edu. any。本模型要求 DNS 能解析一级域名为 Any 的域名，并返回相应的 Anycast 地址。DNS 解析 Anycast 地址的详细过程同其他类型的域名解析过程相同，这里不再赘述。

3.2 客户端

在本模型中，客户端需要增加三个功能：(1)根据域名判断服务类型是否为 Anycast；(2)如果是 Anycast 类型，那么将解析得到的 Anycast 地址以及其自身的 Unicast 地址一起发送到 Anycast 控制器进行解析，并且能够接收 Anycast 控制器返回的最佳 Anycast 组成员的 Unicast 地址，然后同该 Unicast 地址所确定的服务器进行正常的通信；(3)整个通信结束后，统计出本次服务的 TRT，RTT 以及传输的总字节数，并将其封装在一个数据包中发送给 Anycast 控制器，如果 Anycast 最佳成员不可达，则发送出错消息给 Anycast 控制器，然后重复(2)。

由于 DNS 中的域名地址是分层结构的，因此，为了区分 Anycast 域名与其他域名，将 Anycast 域名的一级域名设置为 Any，例如：www. njust. edu. any。所以，客户端只要判断要解析的域名的一级后缀是否为 Any，就可以判断出所要请求的服务类型是否为 Anycast 类型。此外，客户端不仅要知道本地路由器的 Unicast 地址，还要知道与本网络相连的 Anycast 控制器的 Unicast 地址，并且能够从 Anycast 控制器返回的消息中获取最佳 Anycast 组成员的 Unicast 地址。最后，客户端要根据发送数据包的时间戳计算出 RTT，以及统计出整个服务的 TRT 与传输的总字节数，然后按照一定格式封装这些数据，发送给 Anycast 控制器。如果对方服务器不可达，那么发送错误消息给 Anycast 控制器，然后继续向 Anycast 控制器提出 Anycast 地址解析请求，这样，用户就不会因为某个 Anycast 成员的离线或者宕机而被迫放弃服务，从而最大限度地保证为用户提供最优的服务。

3.3 Anycast 控制器

在本模型中，Anycast 控制器需要完成以下功能：(1)接收客户端的 Anycast 地址解析请求；(2)根据客户端的 Unicast 地址与 Anycast 组成员的 Unicast 地址，计算出相应的权值；(3)查找最佳 Anycast 组成员，并将其 Unicast 地址返回给客户端；(4)接收客户端发送的本次服务参数数据包或者错误消息，并根据其参数按照一定的算法更新数据库或者将错误消息中的不可达 Anycast 组成员进行处理；(5)接收新的 Anycast 成员加入消息，并根据其更新数据库。

本模型中，Anycast 控制器维护了一个数据库，记录了 Anycast 服务器的参数表，该表的数据结构如下：

Anycast 地址	对应的 IP 地址	TRT	BW	RTT
AnycastAddri	IP_{i1}			
	IP_{i2}			
	…			
	IP_{in}			
…	…	…	… …	

在本模型中，权值的计算方法如下描述：

IPv6 中的 Anycast 地址是 Unicast 地址空间中的一部分，所以 Unicast 和 Anycast 地址从结构上没有任何区别。IPv6 的地址格式与 IPv4 不同，一个 IPv6 的 IP 地址由 8 个地址节组成，每节包含 16 个地址位，除了 128 位的地址空间，IPv6 还为点对点通信设计了一种具有分级结构的地址，其分级结构划分如下所示。

3	13	8	24	16	64
FP	TLAID	RES	NLAID	SLAID	interface ID

其中，FP 是可聚合全局地址的格式前缀（例如，001 代表单播地址）；TLA ID 为顶级聚合标识符；RES 为将来使用而保留；NLA ID 是次级聚合标识符；SLA ID 是站点级聚合标识符；Interface ID 为接口标识符。IPv6 全局单播地址的分配方式如下：顶级地址聚合机构 TLA（即大的 ISP 或地址管理机构）获得大块地址，负责给次级地址聚合机构 NLA（中小规模 ISP）分配地址，NLA 给站点级地址聚合机构 SLA（子网）和网络用户分配地址。

在本模型中，根据 IPv6 的分级地址结构，将权值分为四类：如果 Anycast 组成员的 Unicast 地址与客户端的 Unicast 地址的 TLA 不同，那么其权值属于第一类，记做 W1；如果 Anycast 组成员的 Unicast 地址与客户端的 Unicast 地址的 TLA 相同，但是，NLA 不同，那么其权值属于第二类，记做 W2；如果 Anycast 组成员的 Unicast 地址与客户端的 Unicast 地址的 TLA、NLA 都相同，但是，SLA 不同，那么其权值属于第三类，记做 W3；如果 Anycast 组成员的 Unicast 地址与客户端的 Unicast 地址的 TLA、NLA、SLA 都相同，那么其权值属于第四类，记做 W4。不难看出，这里 W4＜W3＜W2＜W1。在具体应用时，这 4 个权值的具体设定要根据 Anycast 组成员的分布情况，以及带宽来决定。

当 Anycast 控制器接收到客户端的 Anycast 地址解析请求时，它会查询数据库看是否有相应的表项，如果有，首先根据客户端的 Unicast 地址以及每个 Anycast 组成员的 Unicast 地址计算出相应的权值，然后返回 TRT 值与权值乘积最小的 Anycast 组成员的 Unicast 地址，如果很多记录所对应的该值都相同，就继续比较 BW 与权值乘积之值，如果该值也相同，再比较 RTT 与权值乘积之值，直到选择一个最优成员为止。如果数据库中没有相应的表项，该 Anycast 控制器就向其相邻 Anycast 控制器发送查询消息。为了有效地获得其他 Anycast 控制器的信息，查询消息里设置了两个字段，一个是路径属性字段，此字段记录了查询消息所跨越的所有网络，目的是防止路由查询消息时出现环路；第二个字段就是 TTL 字段，它用来控制查询消息的路由范围，TTL 开始的时候被初始化为该查询消息所能通过的 Hop 最大值，每通过一个网络区域，TTL 的值递减 1。Anycast 控制器查询消息的整个过程可以描述如下：(1) Anycast 控制器把需要查询的 Anycast 地址封装在数据包中，并以多播的形式把此数据包发送给相邻 Anycast 控制器，并启动一个时钟；(2) Anycast 控制器接收到查询消息之后，首先根据查询字段中的路径属性字段判断其是否形成回路，如果没有，那么查询本机数据库看是否有相应的 Anycast 表项，如果有，则获取该表项的内容并将其发送给源 Anycast 控制器，否则，将查询字段中的 TTL 值减去 1，如果此时的 TTL 值不为 0，那么再将本 Anycast 控制器的 Unicast 地址添加到查询消息中的路径属性字段中，最后将该查询消息发送到其相邻 Anycast 控制器；(3) 发送查询消息的 Anycast 控制器在时钟到达之后，根据接收到的应答消息首先选取一个最佳 Anycast 成员的 Unicast 地址返回给客户端，然后根据接收到的应答消息更新自身的数据库，否则，如果 Anycast 没有接收到任何应答消息，那么将向客户端返回一个错误消息。

在查询消息中，TTL 一般设置为 2 或者 3。

当客户端本次服务完成或者失败之后，它要发送一个本次服务参数数据包或者服务失败消息给 Anycast 控制器，如果是失败消息，那么 Anycast 控制器将把失败消息中的 Unicast 地址所对应的 Anycast 成员进行标记，以便以后进一步检测，如果多次检测该 Anycast 成员仍然不可达，就将其删除；如果是服务参数消息，那么就更新自身数据库，具体更新过程描述如下：

$$TRT=\alpha TRT_{old}+(1-\alpha)TRT_{new}$$

这里的 TRT_{old} 指当前数据库中的 TRT 值，TRT_{new} 指客户返回的本次服务的参数值，α 是一个常量，可以根据网络性能的稳定性来决定 α 的取值，本模型取值为 0.25。

$$RTT=\alpha RTT_{old}+(1-\alpha)RTT_{new}$$

这里的 RTT_{old} 指当前数据库中的 RTT 值，RTT_{new} 指客户返回的本次服务的参数值，α 是一个常量，可以根据网络性能的稳定性来决定 α 的取值，本模型中其值与 TRT 相同。

服务的总体时间包括建立连接时间，数据传输时间和关闭连接时间，所以，根据如下公式可以计算出本次服务的平均带宽：

$$BW=S/(TRT-3.5\times RTT);$$

这里的 TRT 指本次服务的总体时间；RTT 指往返时间；S 指传输的数据总量。通过以上公式，可以计算出本次服务的带宽值。

在本模型中，所有的 Anycast 控制器都属于一个 Multicast 地址的组成员。一个 Anycast 组加入新成员时，要发送一个 Multicast 给所有的 Anycast 控制器以便其更新数据库，同时，Anycast 控制器之间也可以利用 Multicast 完成彼此的信息交互。当 Anycast 控制器接收到一个服务器加入 Anycast 组的消息时，它首先对该消息进行认证，在确定安全的前提下，在对应的 Anycast 表项中添加该服务器的 Unicast 地址信息，同时把其他字段的值设置为初始值（即最优值）。

3.4 服务器

如果一个服务器要求加入一个 Anycast 组的时候，它要通过 Multicast 发送一个消息给所有的 Anycast 控制器，通知它们该服务器的当前身份。该消息包括服务器的 Unicast 地址等信息，可以通过在 BGP（域间）或者 IGMP（域内）中增加新的消息类型来实现。这里需要注意的一个问题就是服务器和所有 Anycast 控制器之间的通信要采用一定的安全措施，以防止假冒和恶意攻击。

（下转第 106 页）

序可以进行安全通信。

3.3 P2P 中的密钥交换协议

本系统采用 RSA 密钥交换协议，其安全性建立在大整数分解大素数因子的数学难题上。其安全素数生成算法基于文[8]的下述定理。

定理 1 设整数 F 的素因子分解为 $F=\prod_{i=1}^{s}q_i^{l_i}$，$q_i$ 为素数，l_i 为正整数，$P=2RF+1$，R 为整数；如果存在整数 a，使得

$$\begin{cases} a^{P-1}\equiv 1(\bmod P) \\ \gcd(a^{(P-1)/q_i}-1,P)=1(i=1,2,\cdots,s) \end{cases} \tag{3}$$

则 P 的每个素因子是形如 $mF+1$ 的素数，$m>1$；再者，如果 F 为大于 R 的奇数，或 $F>\sqrt{P}$，则 P 为素数。

4 仿真试验

为了评估本文提出的 P2P 安全通信模型，我们进行了一系列的仿真试验。仿真的应用场景是 P2P 网络中文件共享。试验环境为 5 台 PC(CPU：PIII 1G，RAM：256M，OS：Linux)通过 100M 以太网互联。试验仿真了 2000 个节点的 P2P 网络，其中 CA 中心 5 个，共享文件 5000 个，随机分配到所有节点上。节点对共享文件的请求是随机的，每个用户在仿真过程中平均完成至少 20 次交易。

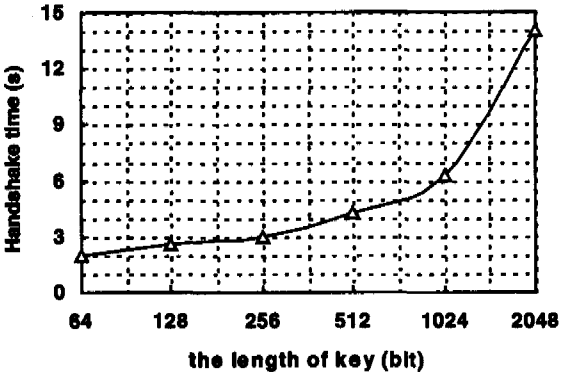

图 4 握手时间随密钥长度的变化

我们仿真了密钥长度在 64～2048 位的证书密钥和密钥交换过程，比较了相应情况下 P2P 的安全握手协议消耗的时间，结果如图 4 所示。

从图 4 中可以看出，密钥长度越长，将会导致握手时间延长，这是因为密钥的加密解密速度，以及 CA 证书在网络中的传送速度，将受密钥长度的影响。密钥越长，其被破解的可能性就越小，安全性也越高，但为之花费的处理时间也越长。仿真试验中，当采用常见的 1024 位密钥时，节点双向认证的握手时间在 6.2 秒。这时对 P2P 服务延迟的影响较小，而 P2P 网络通信安全也获得足够的保证。

结论 本文对 P2P 网络中存在的安全问题进行了分析，提出了一个 P2P 安全通信模型。模型采用分布式的 CA 认证中心，并在节点通信中使用 SSL 协议。通过分析和仿真试验说明，该模型能够保证节点间通信的安全性，达到了设计的目标并具有较好的性能。

参 考 文 献

1 Golle P, Leyton-Brown K, et al. Incentives for Sharing in Peer-to-Peer Networks. In: Proceedings of the ACM Conference on Electronic Commerce, Oct. 2001

2 Detsch A, Gaspary L P, et al. Towards a flexible security framework for peer-to-peer based grid computing. In: Proc. of the 2nd workshop on Middleware for grid computing, Oct. 2004

3 McKean C. Peer-to-Peer Security and Intel's Peer-to-Peer Trusted Library. SANS Institute Information Security, Aug. 2001

4 Bailes J E, Gary F. Managing P2P security. Communications of the ACM, Sep. 2004

5 Gupta V, Gupta S, Chang S. Performance analysis of elliptic curve cryptography for SSL. In: Proceedings of the ACM workshop on Wireless security, Sep. 2002

6 Chadwick D W, Otenko A. The PERMIS X. 509 role based privilege management infrastructure. In: Proceedings of the seventh ACM symposium on Access control models and technologies, California, Jun. 2002

7 Zhou Lidong, Schneider F B, Van Renesse R. A secure distributed online certification authority. ACM Transactions on Computer Systems (TOCS), Nov. 2002

8 Hastad J, Naslund M. The security of all RSA and discrete log bits. Journal of the ACM (JACM), Mar. 2004

(上接第 83 页)

4 系统性能评估

综上所述，本模型不仅解决了在应用层实现 Anycast 技术的网络负载问题，同时也解决了在 IP 层难以实现的采用多种距离度量方式查找最佳 Anycast 成员的问题，以及在客户与服务器之间可能存在的通信问题。最重要的是，本模型解决了在 IP 层存在的扩展性问题，而且 Anycast 成员在地理位置上分布越广泛，其提供服务的性能会越强大。所以，本模型是一个可扩展的、低消耗的 Anycast 通信模型，它根据每个 Anycast 组成员的当前状态，将客户的服务请求均匀地分布到各个成员中去，以便为用户提供最好的服务，同时，本模型还提供了 Anycast 组成员不可达的保护机制，极大限度地保证为客户提供优质服务。

本模型与当前的网络应用程序以及协议都能很好地兼容，因为本模型对路由器没做任何修改，只对 DNS 服务器、客户端的应用程序以及服务器稍加改动。

在本模型中，由于 Anycast 控制器与客户端之间的信息交换可以认为是在本地网络中实现的，因此对整个网络的性能基本上没有任何影响。虽然 Anycast 控制器的查询消息会占用一些网络资源，但是它的使用并不频繁，所以，对网络的主干网也不会有任何的影响。

目前，该模型在 IPv6 的模拟环境下运行良好。

结束语 Anycast 是 Ipv6 的一个新特性，它可以支持许多服务。本文在 IPV6 的模拟环境下，提出了实现 Anycast 服务的一种新的加权模型，用以解决当前在应用层以及 IP 层实现 Anycast 服务所存在的一些问题。Anycast 作为一种新型的通信模式，具有广泛的前景，但是它还存在许多问题，有待进一步探讨和研究。

参 考 文 献

1 Partridge C, Mendez T, Milliken W. Host anycasting service. RFC 1546, 1993

2 Deering S, Hinden R. Internet Protocol Version 6 (Ipv6) specification. RFC 2460, 1998

3 Hinden R, Deerings. IP version 6 addressing architecture. RFC 2373, 1998

4 Hagino J itojun, Ettikan K. An analysis of Ipv6 anycast Internet Draft. Internet Engineering Task Force, 2001

5 JohnSon D, Deering S. Reserved Ipv6 Subnet anycast addresses. RFC2526, 1999

6 Katabi D, Wroclawski J. A framework for scalable global IP-Anycast (GIA). In: Proc. of SIGCOMM, New York: ACM Press, 2000. 3～15

7 Narten T, Nordmark E, Simpson W. Neighbor discovery for IP version 6 (Ipv6), RFC 1970, 1996

8 Huitema C. Routing in the internet. Prentice Hall, 1996