基于 IPv6 的任播路由协议的研究和设计

Research and Design of Routing Protocols for IPv6 Anycast Communication

李 锦 （中国科学院研究生院 北京 100039）
鲁士文 （中国科学院计算技术研究所 北京 100080）

摘要：任播通信是 IPv6 的新特性，可以在 IPv6 网络中实现面向服务的管理和调度路由的功能。由于没有相关的协议标准，甚至对其路由协议尚没有大致统一的意见，段间任播通信尚未实现。为此，在本文中，设计了两种段间任播通信的路由协议，以支持任播导向型通信。

关键词：IPv6 网络管理 拓扑发现 组播 任播

1 任播简介

任播通信是 IPv6 的新特性，可以在 IPv6 网络中提供服务导向型地址分配功能。和现有路由协议不同，任播地址不是由节点的位置决定的，而是由节点所提供的服务类型决定的。甚至，在任播通信中，客户端可以自动获得与特定服务相对应的适当节点，而根本无需知道服务器的确切位置。

和组播地址一样，任播地址也被分配给一组节点（称为任播成员），但与组播通信不同的是，每次通信仅有一个任播成员参与。任播通信的思想是将逻辑的服务提供者与物理的主机设备分离开。例如，任播地址的分配是基于服务类型进行的，这样，网络服务就像是一台逻辑主机一样了。

然而，根据现有的规范，IPv6 的任播通信还有很多问题需要澄清。本文中提供了适于任播通信的一些应用并讨论了任播通信的优点。

基于 IPv6 的任播通信的另一个问题就是，在其规范中并未提供路由协议，而路由协议在任播通信中是不可缺少的。在设计任播路由协议时，需要很好地解决如下这些问题：

1.1 规模问题

每个任播地址的路由项必须独立存储在路由器上。很容易想象，随着任播地址的广泛应用，将会出现路由表爆炸。

1.2 任播成员选取的标准

任播路由要求将一个任播数据包转发到一个适当的任播节点，然而适当的任播节点应该是随着应用的不同而有所变化的。任播路由的选择标准将显著影响任播通信的能力。

基于这些情况，本文设计了段间任播通信的路由协议。

2 任播路由协议设计

通过比较任播通信与单播/组播通信之间的区别，根据现有的单播/组播路由协议提出了一种任播路由协议。任播通信与单播/组播通信之间有很多异同点，因此可以借鉴现有的单播或组播路由协议。表 1 给出了路由协议的三种基本分类：(a) 距离矢量型、(b) 链路状态型、(c) 核心树。

表 1 路由协议的分类

	距离矢量型	链路状态型	核心树
单播	RIPng	OSPFv3	
组播	DVMRP	MOSPF	PIM－SM
任播	ARIP	AOSPF	PIA－SM

由于每种路由协议都各有优缺点，因此针对每种路由协议都设计了相应的任播路由协议，如 (a) 对 RIP 协议的任播扩展 (ARIP)、(b) 对 OSPF 协议的任播扩展 (AOSPF)、(c) 稀疏模式独立协议任播 (PIA－SM)。

任播通信路由协议包括以下三个步骤（见图 1），ARIP 与 AOSPF 的区别主要体现在步骤二上。

步骤一，初始化任播通信成员；

步骤二，组织及更新路由表；

步骤三，转发任播通信数据包。

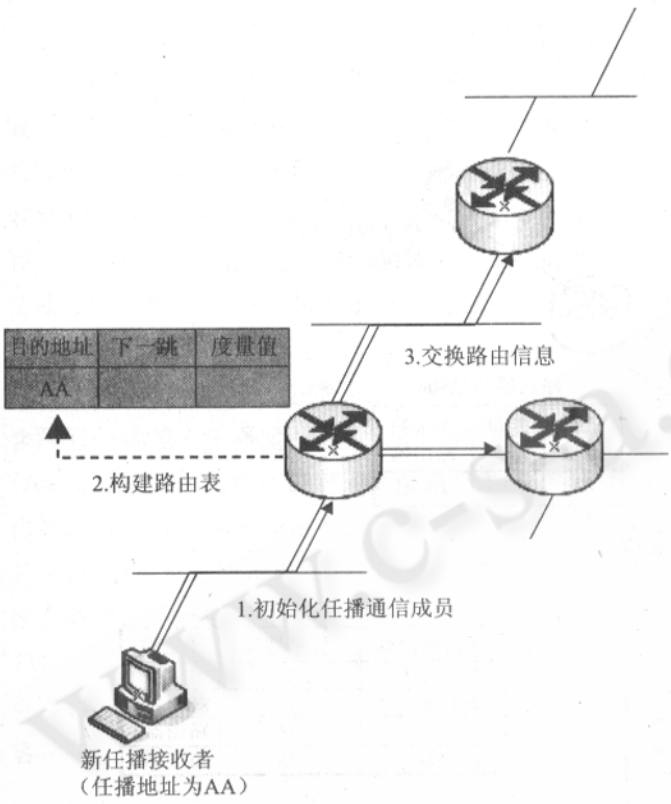

图 1　任播路由协议总览图

任播路由器根据步骤二中形成的路由表来转发任播数据包。值得注意的是，任播路由的步骤三与单播路由完全一样。每台任播路由器仅对单播路由表进行查找，来获得对应于数据包的目标地址项。接下来将对步骤二和步骤三分别进行详细论述。

2.1 初始化任播通信成员组

和组播通信类似，主机加入（或者离开）任播成员组时，必须通知最近的任播路由器其加入（或者离开）的状态。如何发现一台主机（以下称为任播主机）加入任播成员组的方法是不同的，并且必须基于任播主机的位置来选择其方法。如果任播通信的接收者与任播路由器在同一网段，将采用 MLD（组播侦听发现）的扩展版来发现新加入的成员。此方法称为 ARD（任播接收者发现）。任播主机接收到任播路由器发来的 ARD 询问消息后，都会产生一个 ARD 报告消息反馈任播路由器。如果任播主机未接收到 ARD 询问消息，它也会主动发送一个 ARD 报告消息。在任播主机离开成员组之前，会发送一个 ARD 完成消息。因为 ARD 包的目标地址域被设置为链路本地地址中的一个，例如，链路范围所有节点（FF02::1）或者链路范围所有路由器（FF02::2），所以此方法仅适用于主机和路由器在同一网段的情况。

2.2 路由表的组织与更新

如果任播接收方声明的路由项的类型仅仅是接收方的度量值，则路由表的组织更新过程与 ARIP 和 AOSPF 相同，称为声明接收方度量。

2.2.1 声明接收方度量

图 2 显示了当任播路由器仅考虑接收方度量时，路由表的组织及更新情况。当任播路由器仅考虑接收方度量时，它们使用单播路由信息来描述路由拓扑。每个任播接收方就像是任播路由器拓扑树的一个树叶。在详细描述工作过程之前，我们先定义一些相关的路由节点：

- 选定的任播接收方：是同一任播成员组中具有最小度量值的任播接收方。
- 可选的任播接收方：是同一任播成员组中具有次小度量值的任播接收方。
- 选定的任播路由器：是与选定的任播接收方实际或虚拟连接的任播路由器。
- 可选的任播路由器：是与可选的任播接收方实际或虚拟连接的任播路由器。
- 相邻的任播路由器：是实际或虚拟连接的任播路由器。

接下来将描述声明接收方度量的基本操作过程。

（1）通过交换 ARD 询问/报告来发布组成员信息

任播路由器定期地发送 ARD 询问，所有的任播接收方发送 ARD 报告来响应路由器的询问，告知自己的组成员信息及度量值。如果任播接收方没有收到 ARD 询问，它们将主动发送 ARD 报告。收到 ARD 报告后，任播路由器就产生/更新本地 ARDB（Anycast Receiver Database）数据库中的表项信息。ARDB 中的每条表项信息都保存了三部分内容：任播地址、接收方度量值以及任播路由器的单播地址。

（2）发送新任播接收方的信息

任播路由器收到 ARD 报告后，它将向相邻的任播

路由器发送新任播接收方的信息(例如,在 ARDB 中的内容)。

(3) 组织路由表和 ARDB 数据库

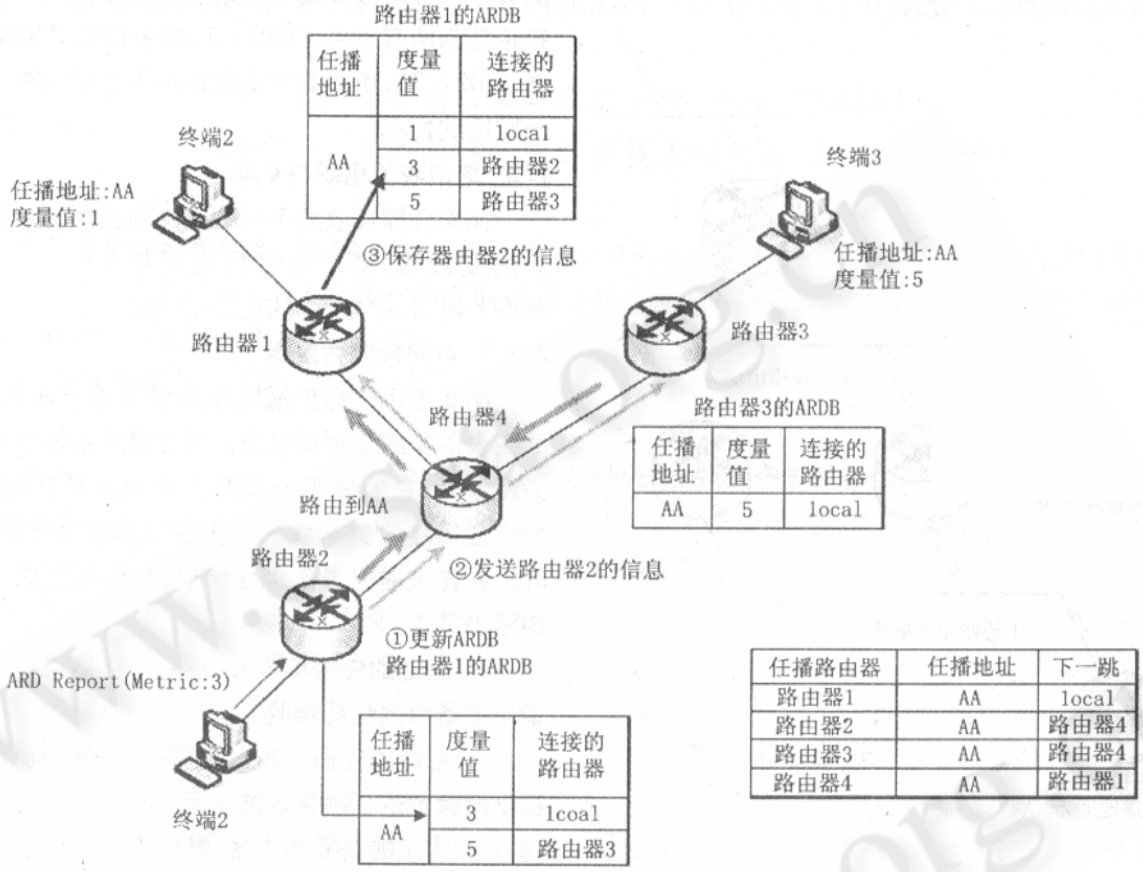

图 2　声明接收方度量信息的基本步骤

收到 ARDB 的表项信息后,任播路由器为 ARD 报告中记载的任播地址查找路由项,并将 ARD 报告中的度量值与所匹配的路由项中的度量值进行比较。如果 ARD 报告中度量值小于路由项中的度量值,则任播路由器会将该度量值改为较小的那个值。然后,任播路由器将向所有的邻居任播路由器发送该表项信息,只有传送 ARD 报告的邻居任播路由器除外。通过一跳一跳地传播 ARD 报告,所有的任播路由器都获得了最小度量值及转发方向。接着,发送到任播地址的所有数据包都被传送给选定的任播接收方。

当 ARD 报告中所附的任播接收方地址与新任播接收方的地址相同时,与任播接收方相连的任播路由器将保存该 ARDB 表项信息。对度量值进行更新时将会用到所保存的 ARDB 表项信息。如果选定的任播路由器没有其它任播接收方的 ARDB 表项信息,或者通过阈值溢出消息得知选定的任播接收方超载,则选定的任播路由器将会发送大量的 ARD 询问消息来寻找其它的任播接收方。此外,如果任播路由器收到此消息,它也不知道其它的任播接收方。然后,每台任播路由器都会各自发送应答消息。这样一来就会浪费大量的通信资源。

因此,如果 ARDB 中的度量值大于 ARD 报告中所附的任播接收方度量值,每台任播路由器就会将收到的表项信息存入 ARDB 数据库。

基本上所有的客户请求都会发送给具有最低度量值的任播接收方(称为选定的任播接收方)。当选定的任播接收方不再具有最低度量值时,将会选择其它的任播接收方(称为可选的任播接收方)作为新的选定任播接收方。为了寻找可选的任播接收方,选定的任播接收方会从 ARDB 的所有表项信息(当前选定的任播接收方除外)中找出最低的度量值。

2.2.2 支持链路度量值 – ARIP

ARIP 和 AOSPF 采用不同的机制来收集链路度量值。由于篇幅限制，本文仅讨论 ARIP。AOSPF 的设计方法与 ARIP 基本相同，只不过是对 OSPF 进行修改。

(1) 通过交换 ARD 询问/报告来发布组成员信息

任播路由器定期地发送 ARD 询问，所有的任播接收方发送 ARD 报告来响应路由器的询问，告知自己的组成员信息(例如，任播地址)。如果任播接收方没有收到 ARD 询问，它们将以特定的时间间隔(例如，每 30 秒)主动发送 ARD 报告。任播路由器的定期更新是由任播接收方发来的 ARD 报告触发的。

(2) 发送 ARI 消息

收到 ARD 报告后，任播路由器会产生一个至少包含两部分内容(任播地址、度量值)的任播路由信息(ARI)。然后，任播路由器将 ARI 发送给相邻的任播路由器。当任播路由器向相邻的任播路由器发送 ARI 消息时，它会用 ARI 消息中的度量值加上输出接口的链路度量值作为新的度量值。这是因为任播接收方与客户端之间的链路度量值非常重要。任播接收方作为服务器，接下来将会有大量的数据从任播接收方传送到客户端。

(3) 接收 ARI 消息，更新路由表和/或锁定列表

任播路由器收到 ARI 消息后，首先检查 ARI 消息中的任播地址是否已保存在路由表中。如果路由表中没有该任播地址，则任播路由器将在路由表中注册该任播地址。然后，任播路由器改写 ARI 消息中的度量值，并将其发送给所有其它相邻任播路由器(发送来该 ARI 消息的方向除外)。此外，任播路由器还会将 ARI 消息中的度量值与现有路由项的度量值进行比较。

收到 ARI 消息的表项信息后，任播路由器就会给 ARI 消息指定的任播地址查找路由项信息，并将 ARI 消息中的度量值与匹配路由项的度量值进行比较。如果 ARI 消息中的度量值小于路由项中的度量值，任播路由器将会用较小的度量值替换原有的度量值。然后，任播路由器向所有其它相邻的任播路由器传送该表项信息，发来 ARI 消息的路由器除外。任播路由器向相邻的任播路由器发送 ARI 消息时，它会用 ARI 消息中的度量值加上输出接口的链路度量值作为新的 ARI 度量值。通过一跳一跳地传播 ARI 消息，所有的任播路由器都获得了最小度量值及其传输方向。接着，发送到任播地址的所有数据包都被传送给选定的任播接收方。

否则，任播路由器将检查接收 ARI 消息的方向。如果现有路由项的输出接口与收到 ARI 消息的路由项不同，就意味着存在不同于选定的任播接收方的其它任播接收方。任播路由器会在锁定列表中保存一对信息(任播地址、度量值)。只有当现有路由项的度量值增加且不再是最小度量值时，才会使用保存在锁定列表中的路由项信息。而且，ARI 消息意味着更新现有的路由项信息。因此，如果现有路由项的度量值增加，任播路由器就可以从锁定列表中选取具有较小度量值的其它路由项。然后，任播路由器就会将这个可选的路由项信息存入路由表，而 0000 将现有的路由项信息存入锁定列表。

3 任播路由协议的实现

3.1 ARD

任播接收方传送度量时，使用了与 MLD 扩展版同样的方法。本文通过扩展 ARD 包格式从任播接收方传送度量信息。度量信息中包括度量类型和度量值。度量类型用于支持任播地址的多种选择标准。度量类型又具体分为两类：接收方度量和链路度量。ARD 报告消息可以包含多套首部和可扩展的值域。如果一个任播接收方具有多个任播地址，它必须向任播路由器发送多个成员信息及度量信息。为了简化消息的数量提高效率，任播接收方可以在一个 ARD 报告消息中设置多个表项。

3.2 ARIP 和 AOSPF

本文通过修改 GNU Zebra 程序来支持任播路由协议。GNU Zebra 是管理基于 TCP/IP 路由协议的免费软件。它支持多种路由协议，其中包括 RIPng 和 OSPFv3。每个路由协议都是通过一个独立的守护程序(例如，ripngd 实现了 RIPng 协议，ospf6d 实现了 OSPFv3 协议)来实现的。每个路由守护程序都要和 zebra 守护程序进行通信，通过它获得来自系统内核的接口信息和路由信息。由于 Zebra 软件采用了多处理技术，可以很容易地进行升级。每个路由协议可以单独进行升级，而不会影响其它的路由协议。

图 3 为总体实现方案。RIPng 和 OSPFv6 被修改成为 ARIP 和 AOSPF，同时支持单播和任播通信。另外，还增加了一段新的程序来处理 ARD 数据包，并管理本地

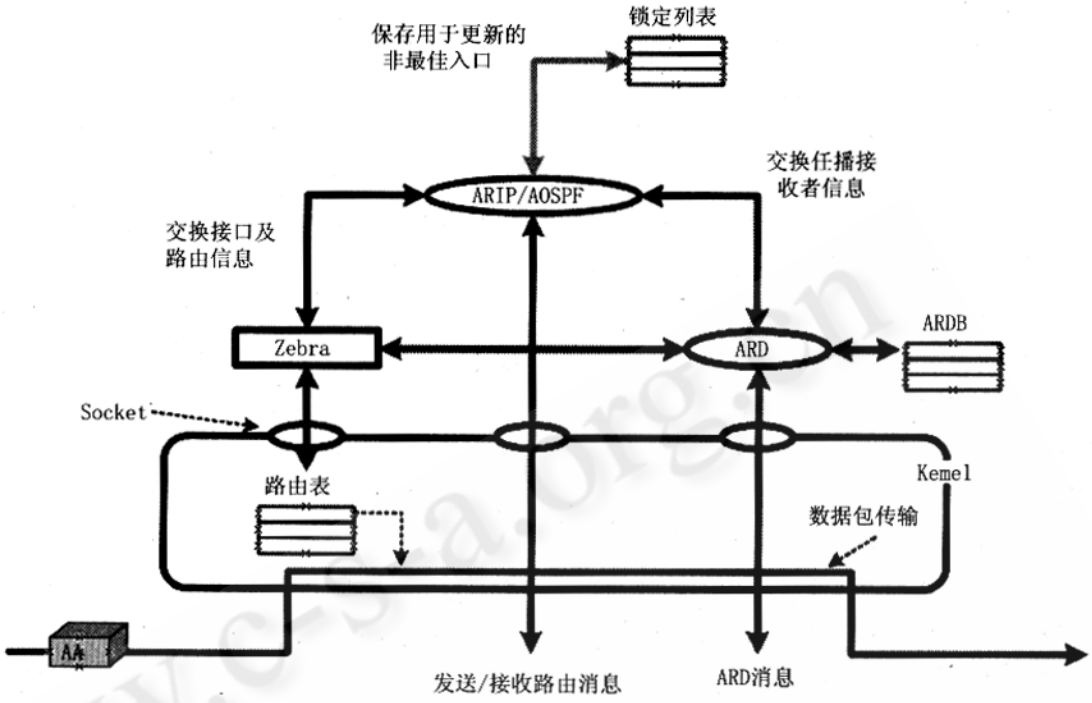

图 3　实现过程

连接的任播接收方。ARIP 和 AOSPF 都要和 ARD 守护程序进行通信，以获得连接到任播路由器上的任播接收方的相关信息。ARIP 和 AOSPF 都使用特殊的消息来处理任播地址（例如，ARI，AM－LSA）和路由信息，以便将任播地址传送给其它的任播路由器。路由守护程序接收到一条路由消息时，首先判断是应该将消息中的信息发送给 Zebra 守护程序，还是应该发送给锁定列表。锁定列表中的路由信息用于更新度量。从路由守护程序或 ARD 守护程序收集到路由信息后，Zebra 守护程序将向位于系统内核部分的路由表添加路由信息或从中删除路由信息。根据 Zebra 守护程序创建的路由表，系统内核来转发数据包。

4　结束语

本文讨论了用于段间任播通信的任播路由控制机制的分析及设计方案，并通过修改现有的路由软件实现了该路由机制。理论分析和实验表明，这种设计方案是可行的。

今后进一步的研究工作将把所设计的协议用于真实的网络中，并对其性能和效率进行深入考察和评估。

参考文献

1　Silvano Gai，IPv6 网络互连与 Cisco 路由器，机械工业出版社，1999.
2　J. Postel，RFC 792，Internet Control Message Protocol，1981.9.
3　D.C. Plu1mmer，RFC826，Ethernet Address Resolution Protocol：On converting network protocol address to 48 bits Ethernet address for transmission on Ethernet hardware，1982.11.
4　S.E. Deering，RFC 1112：Host extensions for IP multicasting，1989.8.
5　A. Conta，S. Deering，RFC 1885，Internet Control Message Protocol（ICMPv6）for the Internet Protocol Version 6（IPv6），1995.12.
6　S. Thomson，C. Huitema，RFC 1886，DNS Extensions to support IP version 6，1995.12.
7　G. Malkin，RFC 2080，RIPng for IPv6，1997.1.
8　R. Hinden，RFC 2373，IP version 6 Addressing Architecture，1998.7.