

TCP/IP 地址结构

单华 范据妮 (铁道科学研究院通信所 100081)

一、TCP/IP 概述

TCP/IP 是多台相同或不同类型的计算机进行信息交换的一组通信协议。它是在七十年代末,八十年代初发展起来的。其层次结构如下表示:

| |
|------------|
| 应用层 |
| 主机 - 主机通信层 |
| 网络接口层 |

最低层是网络接口层(IP),它位于提供通信服务的下层网络之上,为高层提供统一的接口,并实现在互连网络上的地址和路由功能,其下层网络可能是远程网络、卫星链路或 LAN。主机-主机通信层(TCP)提供了互连网络上两个节点进程之间的通信服务。传输的可靠性依赖于这一层协议。应用层与应用服务有关,例如文件传输、虚拟终端和电子邮件等。

TCP/IP 协议有五个明显特征,即可靠性、可互操作性、安全性、灵活性及易于过渡到新协议(如 OSI)的能力。可靠性是由 TCP 提供的。IP 并不保证可靠的数据传送,而只是尽力传递(straight forward)数据包,但是 IP 能够确保数据包不会误传。TCP 提供了可靠的、全双工的、基于连接的数据传送服务。可互操作性是指不同的计算机系统之间可彼此通信,是由三个标准 TCP/IP 协议体现出来。这三个协议是:FTP(文件传送协议)、TELNET(虚拟终端服务)和 SMTP(简单的邮件传输协议)。这些实用程序定义了用户软件与传输层及网络层软件之间的界面,允许不同厂家实现兼容注。TCP/IP 对它的下层支持协议和上层应用协议几乎不作特殊要求。因此,TCP/IP 的使用不受传输介质和网络应用软件的限制。TCP/IP 虽然还需要一些其他协议层辅助,但它可使用原语来允许不同的组合。

二、IP 地址功能及发展过程

如果说操作系统的主要功能是为同一对象定义不

同的名字,这样它可以忙于跟踪这些不同名字的关系的话,网络协议有某些相同的特点。在 TCP/IP 中,有几种不同的方法来指代同一事物。在人机界面上,用字符串“名字”来标识网络,主机和服务。主机名被翻译成 32 位的网络地址来指示主机连接的网络及主机在网络中的位置。服务名被翻译成“端口标识”(TCP 中是一个 16 位的值)。最后地址被映射成“路由”,其中路由是一个数据报要经过的节点序列。

IP 地址是由四个字节组成的。最初设计 IP 地址时,把四字节的 IP 地址分为两个部分。第一部分是网络号,第二部分是主机号。由于当时的网际很小,因此只把网络部分定为一个字节。在网关中保存一个 256 条入口的表,并为本网关直接相连的网络保存一个表。这样路由表目就可以静态设定了。但是这也造成了许多问题,比如若网络或网关移动或发生故障,就会造成混乱,而且 256 个网络也满足不了需求。由于这些缺点,人们对地址的表示方法作了改进,其结构如下:

| 类型 | 高位比特 | 格式 |
|----|------|-------------------|
| A | 0 | 7 比特网号, 24 比特主机号 |
| B | 10 | 14 比特网号, 16 比特主机号 |
| C | 110 | 21 比特网号, 8 比特主机号 |

这种 IP 地址是自标识的,即由最高的几个比特标识网络部分及主机部分所占的比特数。A 类地址主要满足大规模网络的情况,其最大允许的网络数为 $2 * * 7 - 3$ (全 0, 全 1 和 127 除外),但在每一个网络中,它允许有较多的主机数,主机数的最大值为 $2 * * 24 - 2$ (全 0 和全 1 除外)。B 类地址主要满足中等规模网络的情况,其最大允许的网络数为 $2 * * 14 - 2$,最大允许的主机数为 $2 * * 16 - 2$ 。C 类地址主要满足小规模网络的情况,其最大允许的网络数为 $2 * * 21 - 2$,最大允许的主机数为 $2 * * 8 - 2$ 。当网络地址或主机地址部分出现全 0 或全 1 时,IP 对其作特殊处理,一般情况下,全 1 代表所有的,全 0 代表本地。

尽管这种两层结构即简单又有效,一些组织仍然发现其不够完善并在两层的基础上又增加了第三层——子网。三层模式对中等规模的组织(如:拥有多于一座建筑的大学或公司)是很有用处的。这些组织需要多于一个的 LAN 来覆盖“本地地域”。为这些组织分配一个网络地址并把这个网络分割成子网,给每个建筑分配一个子网号。在组织内部,各个建筑的位置都是显式指明的。对组织外部,只指明整个组织的位置。这样即增强了组织的整体性,也节约了资源。在子网出现的初期,人们使用的是定长子网,也就是说对一个网络而言,所有的子网号长度是一致的,全网用统一的子网掩码。例如:给定网络 A 分配网号 129.0,把 A 分为 16 个子网。地址分配如下表示:

| | | | |
|----------|----------|----------|----------|
| 10000001 | 00000000 | xxxxhxxx | hhhhhhhh |
|----------|----------|----------|----------|

x—子网号, h—主机号

这种分割子网的方法的优点是简单,缺点则是对地址空间的利用率不高。因为一个组织内部的各个子网大小不一,采用统一长度的子网编码必然为所有子网分配最大的子网空间,对那些小网来说无疑是一种浪费。于是人们又提出了变长子网的概念,在这种分割方式中,网络被分割成不同大小的子网。例如:给定网络 A (129.0.1),把 A 分割成容量为 126 个主机的子网 S1 和容量为 62 个主机的子网 S2, S3。子网编码方案如下:

| | | | | |
|-----|----------|----------|----------|----------|
| S1: | 11000000 | 00000000 | 00000001 | 0hhhhhhh |
| S2: | 11000000 | 00000000 | 00000001 | 10hhhhhh |
| S3: | 11000000 | 00000000 | 00000001 | 11hhhhhh |

随着 Internet 的发展,又出现了新的问题,一方面, B 类地址空间即将耗尽;另一方面,路由表增长过速。造成这种局面的原因在于,对很多组织, C 类网络太小, B 类网络过大,但是由于 IP 地址置标识的特点,只好分配 B 类地址,在路由表中,若为每个网络保存一个入口,根据前面的分析我们知道,仅 C 类网络就要 $2 * 21 - 2$ 条表目,这显然是不可行的。为了解决这些紧迫的问题,保证 Internet 的正常运转,人们提出了无类型的地址分配方案。

所谓无类型地址就是说不再区分 A, B, C 类地址,而是将整个地址空间分为若干区间,每个区间用一个地址

掩码对来表示。如(129.0.0.0, 255.255.128.0):

| | | | |
|----------|----------|----------|----------|
| 10000001 | 00000000 | 0hhhhhhh | hhhhhhhh |
|----------|----------|----------|----------|

从 129.0.0.1 到 129.0.127.254 共 $2 * 15 - 2$ 个主机地址的空间。这样四个字节的 IP 地址空间就可以灵活分配。而边界路由器则可以进行路由总结,节约路由表占用的内存资源。

这种无类型地址只解决了短期问题,从现在 Internet 的增长速度来推算,不久的将来,四个字节的 IP 地址将全部耗尽。因此人们正在致力于新版本 IP 协议制订。1995 年 12 月推出的第六版 IP (IPv6) 中, IP 地址扩展成 16 个字节,分为三种类型: Unicast (单点), Anycast (任意点) 和 Multicast (多点)。其中 Unicast 是单一接口的标识,目的地为 Unicast 的数据报被送到指定接口。Anycast 与 Multicast 都是一组接口的标识,不同的是,目的地为 Anycast 的数据报被送到 Anycast 指定接口组中最近的一个,而目的地为 Multicast 的数据报被送到指定接口组中的所有接口。IP 地址的类型是由高位标识,由这些高位标识组成的变长域称为格式前缀 (Format Prefix - Fp)。例如:多点地址 (Multicast address) 的格式前缀为 1111 1111, 供应商单点地址 (Provider - Based Unicast Address) 的格式前缀为 010。目前,整个地址空间的 15% 被初步分配出去了,剩下的 85% 保留给将来使用。这种新的地址分配方案为各种地址类型定义了很多形式,从而大大增加了 IP 地址的灵活性。例如:一种 LAN 环境下的 Unicast 形式如下所示:

| | | |
|------|-----------|-------|
| n 比特 | 80 - n 比特 | 48 比特 |
| 前缀 | 子网标识 | 接口标识 |

这种形式使得自动配置 IP 地址非常简单。一个节点可以通过倾听路由器发送的路由信息来发现子网标识,然后利用它自身独一无二的 IEEE MAC 地址构造 IP 地址。

总之, IP 地址从产生到现在经历了一系列的从概念到实现的变化,这些变化都是为了适应变化了的环境,适应不断扩大的 Internet 而出现的。相信在不久的将来,又会出现一些新的概念,出现一些新的地址分配方案,但是我们深信,所有的事物都有其自己的规律,了解 IP 地址的过去和现在将帮助我们更好地创造未来。

(来稿时间:1997 年 2 月)