

# IP 子网物理拓扑结构发现研究

李学锋, 杨健强

LI Xue-feng, YANG Jian-qiang

襄樊学院 计算机教学中心, 湖北 襄樊 441053

Computer Teaching Center, Xiangfan University, Xiangfan, Hubei 441053, China

E-mail: to\_lxf@yahoo.com.cn

LI Xue-feng, YANG Jian-qiang. Research on physical topology discovery algorithm for IP subnets. *Computer Engineering and Applications*, 2009, 45(1): 131-133.

**Abstract:** In this paper, the topology of a IP subnet is abstracted to a tree and the MAC addresses in the AFTs (address forwarding table) of interconnection switches are analyzed. A set of theorems to determine the switches' relationships is also presented. Based on the above theorems, a new top-down algorithm for physical topology discovery is proposed, which can construct the whole topology utilizing the AFTs. Compared with other approaches to Ethernet topology discovery, this algorithm is more efficient and practical.

**Key words:** network management; physical topology discovery; IP subnet; address forwarding table

**摘要:** 将 IP 子网的拓扑结构看做一棵拓扑树, 对互连的交换机的地址转发表中的交换机的 MAC 地址进行了分析, 给出一组判定定理用以确定交换机间的连接关系。基于上述定理, 提出了一个新的自顶向下的拓扑发现算法, 该算法能够利用地址转发表构造出整个网络拓扑结构。与已有的物理拓扑发现算法相比, 该算法具有高效、实用的优点。

**关键词:** 网络管理; 物理拓扑发现; IP 子网; 地址转发表

**DOI:** 10.3778/j.issn.1002-8331.2009.01.041 **文章编号:** 1002-8331(2009)01-0131-03 **文献标识码:** A **中图分类号:** TP393

## 1 引言

随着网络技术的广泛应用, 网络规模的不断扩大, 对网络的有效管理显得日益重要。准确及时的网络拓扑结构信息是网络性能监测与评估、故障发现与定位、资源分配与管理等一系列网络管理工作的基础<sup>[1-2]</sup>。因而找出网络设备之间的互连关系, 即网络拓扑结构, 便成为完成许多重要网络管理任务的前提。

目前网络拓扑发现主要分为网络层的拓扑发现(又称为逻辑网络拓扑)和数据链路层的拓扑发现(又称为物理网络拓扑)两类。逻辑网络拓扑发现是指发现网络层(对应 OSI 的第 3 层)设备及其它它们之间的连接关系, 即路由器到路由器、路由器接口到子网的连接关系。逻辑拓扑发现已经形成了一种成熟通用的算法, 即通过访问路由器每个接口的路由表进行广度优先遍历, 从而发现所有路由器及其所连子网。物理拓扑结构发现是指发现数据链路层设备之间的连接关系, 也即交换机端口之间的连接关系, 以及交换机端口和主机之间的连接关系。物理拓扑算法的研究却因为交换机的转发透明性尚未有公认的算法形成<sup>[3]</sup>。

## 2 物理拓扑算法的相关研究

IETF 于 2000 年推出物理拓扑 MIB (Management Information Base)<sup>[4]</sup>, 试图解决网络层以下拓扑结构的发现问题。但是由于没有确定获取这些 MIB 对象的机制, 关于网络物理拓扑结构的自动发现还有待更多的研究。

2000 年, 贝尔实验室的 Breitbart 等人提出了基于完整地址转发表 AFT (Address Forwarding Table) 的单一网物理拓扑发现方法<sup>[5]</sup>。同时还提出了解决转发表完整性的两个方法: 通过向网络中的所有设备发送伪造源地址为交换机的 ICMP 请求响应包或是减少直接连接定理的苛刻度。然而, 由于转发表的时效性以及目标网络的监控性, 这些方法不能从根本上解决完整性问题。2002 年, 郑海等人提出了一种改进方法<sup>[6]</sup>, 该算法将交换机分为一个上行接口和多个下行接口, 只要保证下行接口 MAC 地址转发表是完整的就可以构造出交换机之间的连接关系。

2005 年, Son 等人提出一个基于生成树协议 STP (Spanning Tree Protocol) 的方法<sup>[6]</sup>, 其基本思想是利用生成树的信息构建网桥设备之间的连接关系。该方法的优点在于时间和空间消耗

**基金项目:** 湖北省教育厅科学技术研究项目 (the Science and Technologies Plan of Department of Education of Hubei Province, China under Grant No. B200625002)。

**作者简介:** 李学锋 (1968-), 男, 襄樊学院计算机教学中心讲师, 主要从事计算机网络方向研究; 杨健强, 男, 襄樊学院计算机教学中心副教授, 主要从事计算机网络、信息安全方向的研究。

**收稿日期:** 2007-12-27 **修回日期:** 2008-08-13

较小,可以发现局域网中的备用链路,缺点是许多交换机不支持生成树协议,适用范围受到一定限制,并且该方法不能发现交换机和主机之间的连接关系。李涛等人在更早时提出了类似的方法<sup>[7]</sup>,该方法没有对备用链路的发现问题加以讨论。

除了上述主流算法外,邱林等人还提出了基于接口流量的接口连接定理<sup>[8]</sup>。但它存在着其他接口及流量误差的干扰,算法实用性不强,且没有解决哑交换机存在的情况。

提出一个新算法利用非完整的地址转发表信息建立 IP 子网的物理拓扑关系,算法把整个网络拓扑看做一棵拓扑树,利用地址转发表信息自顶向下地快速构造出这棵树出来。

### 3 物理拓扑结构发现算法

#### 3.1 模型和理论基础

把管理区域设定为一个 IP 子网,IP 子网定义为任意两个节点不经过路由器通信的最大的 IP 地址集合,即为一个广播域(broadcast domain)。为了避免广播风暴的发生,一个 IP 子网的拓扑结构必须是树型的,因此也称 IP 子网的拓扑结构为拓扑树,生成树协议能够保证 IP 子网的拓扑是一个树状结构。

将管理域抽象为一棵树  $T$ ,管理域中的所有网络设备(交换机和主机)组成了  $T$  的节点集  $D$ ,  $S$  和  $H$  是  $D$  的子集,分别表示节点中的交换机集合和主机集合,  $D=S+H$ 。不同设备端口之间的物理连接作为  $T$  的边  $E$ 。

**定义 1** 在树  $T$  中第  $i$  台交换机的第  $j$  个端口记录为  $S_{ij}$ 。

**定义 2**  $A_{ij}$  为一个 MAC 地址集合,元素均为交换机  $S_i$  的地址转发表中通过端口  $S_{ij}$  收到的数据帧的源 MAC 地址。

**定义 3**  $A_i$  为通过交换机  $S_i$  的下行端口的 MAC 地址集合。

**定义 4** 若交换机端口  $S_{ij}$  的  $A_{ij}$  中未出现其他交换机 MAC 地址,则称端口  $S_{ij}$  为叶端口。若交换机的所有下行端口都是叶端口,此交换机为叶交换机。

**定义 5** 管理结点:当算法运行的主机在欲发现物理网络拓扑的子网中时,将此主机命名为管理结点,若不在,则将目标子网中能转发算法运行的主机发出的数据包的路由器结点定为管理结点。

**定义 6** 上行端口:指端口对应的地址转发表中出现管理结点 MAC 地址的端口。

**定义 7** 下行端口:指端口对应的地址转发表中没有出现管理结点 MAC 地址的端口。

节点的祖孙关系或直系关系是指当一个节点要通过另一个节点才能到达根节点时,这二个节点便是一种祖孙关系或直系关系。

如果说端口  $S_{ij}$  是完全的,是指在给定子网中若交换机  $S_k$  发出的数据帧可以通过端口  $S_{ij}$  到达  $S_i$ ,则  $S_k$  的 MAC 地址必出现在  $A_{ij}$  中。

**定理 1** 如果  $S_j$  的 MAC 地址在  $A_i$  中,那么  $S_i$  一定是  $S_j$  的祖先节点。

**定理 2** 在树型结构的网络中,当所有交换机的下行端口是完全的时候,父交换机  $S_i$  的  $A_i$  中所具有的交换机的 MAC 地址的数目大于它的任何一个子交换机  $S_j$  的  $A_j$  中所含有的交换机 MAC 地址的个数。

**证明:** 设交换机  $S_j$  直接连接在交换机  $S_i$  的  $S_{ij}$  端口上,即  $S_j$  是  $S_i$  的父节点。由于所有交换机的下行端口是完全的,那么  $A_{ij}=A_j+S_j$  的 MAC,所以  $A_{ij}>A_j$ ,由此可知,交换机任何一个端口

$S_{in}$  的  $A_{in}$  都大于其子交换机  $S_n$  的  $A_n$ 。因为  $A_i=\sum_{n=1}^N A_{in}$ ,所以父交换机  $S_i$  的  $A_i$  中所具有的交换机的 MAC 地址的数目大于它的任何一个子交换机  $S_j$  的  $A_j$  中所含有的交换机 MAC 地址的个数。

**定理 3** 在树型结构的网络中,当所有交换机的下行端口是完全的时候,下行端口的地址转发表中包含交换机 MAC 数目最多的交换机是根交换机。

#### 3.2 算法思想

物理拓扑结构的快速发现算法的基本思想是:将整个子网的交换机作为一个集合,并使它们的下行端口是完全的,根据定理 3 找出其根交换机,然后对根交换机的各下行端口进行先序遍历,遍历到每个端口时,以地址转发表中,MAC 地址与此端口相关联的交换机为集合,根据定理 3 找出此集合的根交换机,此交换机即是直接与此端口相连的交换机。然后再对此交换机的端口进行遍历。这样,可以很快地从根开始建立起整个子网的物理拓扑结构。

#### 3.3 活动节点的发现与其类型的确定

拓扑发现的第一步是发现管理域内的所有节点,确定节点类型,生成设备列表。

首先向子网中所有的 IP 地址用异步的方式发送 ICMP 回应请求包,回应的 ICMP 报文如果包含在输入的 IP 中,则该 IP 是一台活动的节点,将这些节点加入活动节点列表中。

然后直接向每一个活动节点使用 SNMP Get 方法得到子网内活动节点信息,通过标准 SNMP MIB 对象 system.sysServices<sup>[9]</sup>确定一个节点的类型。如果 system.sysServices 的第 2 个 bit 位是 1,则该节点是一个交换机,否则认定该节点为一个主机。如果节点不支持 SNMP,也被认定为是一个主机。

#### 3.4 所有交换机的下行端口地址完全性的实现

物理拓扑结构的快速发现算法主要是依据定理 3 来发现根交换机的,定理 3 适用的前提是所有交换机的下行端口是完全的,如何使所有交换机的下行端口完全,便成为算法的关键之所在。根据交换机的工作原理可知,从管理机向所有的交换机发送 ICMP 操作后,就可使所有交换机的端口是完全的。

#### 3.5 哑交换机与集线器的发现

有个问题需要考虑:如果网络中存在不支持 SNMP 的交换机,或者交换机之间通过 HUB 相连,这时只用定理 3,生成的物理网络拓扑就不对了。

完整的树结构如图 1 所示,若所有交换机都支持 SNMP,并且下行端口是完全的,根据定理 3,整个树的根为  $S_1$ ,与  $S_1$  相关联的交换机有  $S_2, S_4, S_5$  和  $S_6$ ,可从中找出下行端口中包含交换机 MAC 地址最多的交换机  $S_2$ 。与  $S_2$  相关联的交换机有  $S_4, S_6$  和  $S_7$ ,再根据定理 3 可以找出  $S_4$  是  $S_4, S_6$  和  $S_7$  的根。这样通过对交换机的各个端口进行遍历可以发现整个树的结构。

如果网络中存在哑交换机或集线器,那么直接用定理 3 就不能生成正确的网络拓扑结构。如果图 1 中的  $S_2$  是哑交换机或集线器,没有被发现,直接用定理 3 生成的拓扑图就如图 2 所示。

当考虑网络中可能存在哑交换机和集线器时,就必须要对定理 3 进行扩展:

设与交换机  $S_i$  的一个端口  $S_{ij}$  相关联的交换机的 MAC 地

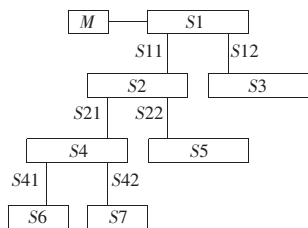


图1 完整的树结构

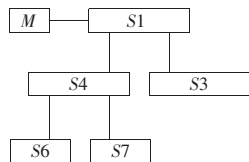


图2 没发现S2的树结构

址的集合为集合  $A$ , 集合  $A$  中的元素的个数为  $|A|$ , 即与端口  $S_j$  相关联的子交换机集合包括的交换机的个数为  $|A|$ , 根据定理 3, 找出此集合中下行端口中包含交换机 MAC 地址最多的交换机  $S_m$ , 设交换机  $S_m$  的下行端口中包含交换机 MAC 地址的个数为  $N$ :

若  $N=|A|-1$  时, 则此交换机  $S_m$  就为此子交换机集合的根交换机, 其直接与根交换机  $S_j$  的  $S_j$  口直接相连;

若  $N<|A|-1$ , 则表示存在一个哑交换机或集线器与  $S_j$  直接相连。集合  $A_m+S_m$  中的交换机以  $S_m$  为根交换机, 连在哑交换机上。然后根据定理 3, 集合  $A=A-A_m-S_m$  的 MAC, 找出集合  $A$  中的根  $S_n$ , 若  $S_n$  的  $A_n$  中的交换机 MAC 的个数等于  $|A|-1$ , 则集合  $A$  中的交换机以  $S_n$  为根直接连到哑交换机的下一个端口上; 若小于  $|A|-1$ , 则说明集合  $A_n+S_n$  中的交换机以  $S_n$  为根直接连接在哑交换机的下一个端口上, 然后令  $A=A-A_n-S_n$  的 MAC, 重复上面的操作。

### 3.6 算法的具体描述

(1) 确定子网内的活动节点及节点的类型, 建立交换机列表: 异步 ping 子网内所有的节点, 确定活动节点, 对活动节点使用 SNMP Get 方法获取 system.sysServices, 根据 system.sysServices 的值以及对 SNMP 的支持情况, 判断节点的是交换机还是主机, 建立子网内的所有交换机列表。

(2) 依次读取每台交换机的地址转发表。

(3) 从各交换机地址转发表中构造每台交换机的上行端口与下行端口集合, 并统计出每台交换机的下行端口集合中包含的其他交换机的 MAC 地址的个数。

(4) 以管理机作为一个交换机, 并将其放入交换机队列, 以上面发现的所有的交换机 ( $N$  个) 作为管理机的下行端口中的交换机集合的元素, 并设管理机只有一个下行端口。

(5) 从交换机队列中, 取出一个交换机, 为了描述方便, 设此交换机为  $S_j$ , 并以  $S_j$  的第一个下行端口为当前端口。

(6) 以当前端口中转发表中的交换机为集合  $A$ , 设集合  $A$  的元素个数为  $|A|$ , 求出此集合中的下行端口中包含交换机 MAC 地址最多的交换机  $S_m$ , 若  $S_m$  的下行端口中的交换机个数为  $|A|-1$ , 则说明交换机  $S_m$  直接连接在  $S_j$  的当前端口上。将交换机  $S_m$  放入交换机队列, 转(8)。

若交换机  $S_m$  的下行端口中的交换机的数目小于  $N-1$ , 说明  $S_j$  的当前端口直接连接在一个哑交换机或非管理的集线器上,  $S_j$  的当前端口与此哑交换机的上行端口相连, 且交换机  $S_m$  也直接连接在哑交换机上, 并设其连接在哑交换机的下行端口 1 上。将  $S_m$  放入交换机队列。集合  $A=A-S_m$  的 MAC- $A_m$ 。

(7) 找出集合  $A$  中的下行端口包含交换机 MAC 最多的交换机, 若此交换机的下行端口的交换机的 MAC 的数目为  $|A|-1$ , 则表明, 此交换机是剩下的交换机的根, 并且此交换机直接连接在哑交换机的下一个下行端口上, 将此交换机放入交换机

队列。转(8)。若此交换机的下行端口中的交换机的数目小于  $|A|-1$ , 表明此交换机直接连接在哑交换机的下一个下行端口上, 将此交换机放入交换机队列。集合  $A=A$ -此交换机的 MAC-此交换机下行端口中的交换机 MAC。转(7)。

(8) 若  $S_j$  的下行端口还没有访问完, 以  $S_j$  的下一个端口为当前端口, 转(6); 若交换机队列不空, 转(5)。

(9) 当交换机队列中的交换机都被处理完毕时, 算法结束, 此时 IP 子网的物理拓扑也已经建立起来了。

### 3.7 与其他算法的比较

目前已经有多种物理网络拓拓扑算法。文献[1]提出的方法要求交换机的地址表是完整的, 即交换机的地址转发表中包含所有其它的交换机 MAC 地址, 由于在实现地址转发表的完全时, 比较麻烦与费时, 因此具有很大的局限性。文献[5]提出一种可以利用不完整的地址转发表自下而上地(从叶子向根)来发现物理拓扑, 但没有解决网络存在哑交换机与集线器的问题。文献[10]对文献[5]的算法进行了扩展, 使其能发现哑交换机, 但提出的方法需要对树进行重新遍历, 操作比较复杂。

为了检验算法的特性, 利用 VC++ 与 SNMP++ 实现算法, 并在如图 1 的网络结构中进行了运行, 能正确地发现网络拓扑。能很好地发现哑交换机和集线器, 可以一次性地建立起交换机之间、交换机与主机的物理拓扑结构。

将本文算法与其它的一般 Ping 程序算法在多种结点数目情况下的检测用时进行了实验, 结果如图 3 所示。利用本文算法时, 一个 C 类网可以在几秒内检测完毕, 极大地提高了算法的性能和效率。

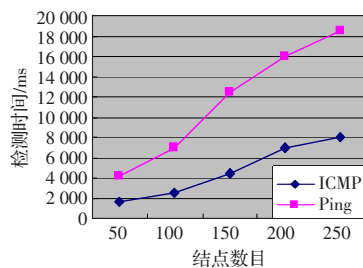


图3 检测时间的比较

## 4 结论

对 IP 子网的物理结构进行了分析, 并基于标准的 SNMP、MIB 协议中一些相关信息, 提出并证明了可用于网络拓扑发现的定理。并以此为基础, 提出了一种新的自顶向下的 IP 子网物理拓扑自动发现算法。该算法在建立网络拓扑时, 充分利用了网络拓扑的树状结构特点, 能有效地减少拓扑发现的时间, 并准确发现 IP 子网的物理拓扑结构。

## 参考文献:

- [1] Breitbart Y, Garofalakis M, Martin C, et al. Topology discovery in heterogeneous IP networks[C]//Proc of INFOCOM 2000. New York: IEEE Press, 2000: 265-274.
- [2] Lowekamp B, OpHallaron D R, Gross T R. Topology discovery for large ethernet network networks[C]//SIGCOMM p01, August 2001.
- [3] 石玫, 李祥和. 基于 STP 的物理拓扑发现算法研究[J]. 计算机工程与应用, 2007, 43(9): 148-150.
- [4] Bierman A, Jones K. RFC 2922 Physical Topology MIB[S]. 2000.