第 33 卷 第 11 期 计 算 机 工 程 2007 年 6 月
Vol.33 No.11 Computer Engineering June 2007
·网络与通信· 文章编号：1000—3428(2007)11—0103—02 文献标识码：A 中图分类号：TP393.07

异构以太网物理拓扑发现的简单算法

杨 婷，裴喜春，周根宝
（内蒙古农业大学计算机与信息工程学院，呼和浩特 010018）

摘 要：针对网络由不同的网络产品所构成的情况，给出了一种基于 SNMP 协议的物理拓扑发现算法，依据现有的理论得出结论。该算法以简单的方式，完整、高效地进行了物理拓扑发现，在子网中可以检测到 HUB 和不支持 SNMP 协议的交换机。
关键词：异构以太网；物理网络拓扑；SNMP MIB

Simple Algorithm of Physical Topology Discovery in Heterogeneous Ethernet

YANG Ting, PEI Xichun, ZHOU Genbao
(College of Computer and Information Engineering, Inner Mogolia Agricultural University, Hohhot 010018)

【Abstract】 This paper describes a algorithm of physical topology which is made up of elements of multi-vendor, which bases on standard SNMP protocol. A theorem is educed in terms of theories of seniorities and practice. Based on this theorem a kind of better algorithm is proposed, which can discover physical layer topology in a simple, complete and efficient way. The algorithm can be applied to discover hubs and bridges that don't support SNMP protocol in subnet.
【Key words】 Heterogeneous ethernet; Physical network topology; SNMP MIB

1 概述

利用网络拓扑可以实现许多网络管理任务，比如提前规划或者动态地管理网络资源，进行服务器定位、故障分析等。目前网络拓扑发现主要分为逻辑网络拓扑(网络层的拓扑)和物理网络拓扑(数据链路层的拓扑)这两类。很多厂家，例如 HP的 Open View 网络结点管理系统、IBM的Tivoli等系统已经加入了自动发现网络层拓扑的功能，但大部分厂家的产品不能获得链路层中各设备之间复杂的连接关系。而随着交换技术的发展，交换设备被大量引入，急需一种能够完整、有效地发现物理网络的拓扑算法。然而发现物理网络拓扑的困难在于第 2 层网络设备，没有信息明确地表述相互的连接关系。交换机在IP层对于终端来讲是透明设备，传统的逻辑层拓扑发现机制无法获知交换机间的连接信息。由于网络中可能存在多个厂商的设备，拓扑发现算法必须具有通用性，因此，不能依赖某个厂商私有的协议或指定的数据。发现以太网中的物理拓扑是一件很难完成的任务。然而，简单网络管理协议SNMP提供了从交换机中获取信息的统一接口，并且定义了一系列标准的管理信息库，其中包括MIB-Ⅱ[1]和 Bridge MIB[2]。通过访问这两个管理信息库，可以获知交换机的系统属性和转发表。其中转发表是用于指导交换机对到达的数据包进行转发的依据。本文依据标准的SNMP[3]、相关MIB信息、定理，以简单的方式实现异构网络环境下的物理拓扑发现。

在描述算法之前，本文给出一些说明，因为现实中以太网之间是通过路由器相连的。为了使算法具有实用性，搭建实验环境如图 1 所示，其中包含路由器。在算法中增加了对交换机和路由器的判断(第 2 节给出了详细的叙述)。由于该算法是基于生成树之上的，因此可以不考虑那些非激活状态的边(这些边由生成树协议——STP 删掉)。为了更好地说明问题，本文算法只包含路由器(R)、交换机(S)，不包含 PC 机。

2 子网中路由器和交换机的判定

从图 1 可以看到，要想得到物理拓扑，需要知道交换机与交换机端口之间的连接关系，还要确定交换机与路由器端口之间的连接关系(由于所搭建的实验环境包括一台路由器)。要想了解这些情况就要解决以下问题：子网内各交换机的 IP 地址，路由器端口地址，如何区分路由器和交换机。

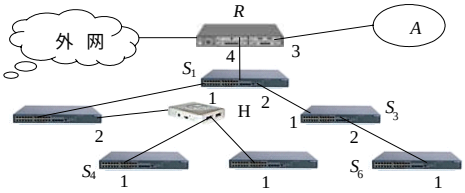

图 1 物理拓扑

(1)网管站 A(参见图 1)向子网中的所有设备按子网地址依次加 1 的方式发送 ICMP 包，当相应设备发回应应答时，可确定设备存在且处于激活状态，从而获得了该活动设备的 IP 地址；若无回应，避免其他原因造成回应请求及回应包丢失，还要发第 2 次、第 3 次，才能确定有无该设备。值得注意的是，为了提高整体拓扑发现的速度，本文采用的是异步的方式发送 icmp 包。

(2)从所有活动的设备当中，区别出路由器端口地址和交换机，逐个对子网内的网络地址发送 GetRequest 请求，请求

作者简介：杨 婷(1979－)，女，讲师，主研方向：计算机网络；裴喜春，教授、博导；周根宝，教授
收稿日期：2006-11-03 **E-mail：**yangting@imau.edu.cn

的对象是 mib 库中 sysServices 变量，对应的 oid=1.3.6.1.2.1.1.7.0，收到 GetResponse 响应报文后，按 sysServices 值来判断响应报文是否来自交换机设备。因为 sysServices 对象的取值按 7 位编码来解释。该数值可表示为

$$\text{sysServices}=\sum_{i=1}^{Li}2^{\text{Li-1}} \quad (1)$$

如果是交换机，它工作在第 2 层，根据式(1)，sysServices 值为 2^2-1=3，由于路由器提供链路层和网络层服务,因此它的 sysServices= $2^{2-1}+2^{3-1}$=6。

3 算法理论依据

Bell实验室的Yuri Breitbart等人提出了各自基于地址转发表的算法[4]，但该算法要求保证交换机的地址转发表是完整的。这在现实中是很难实现的。文献[5]给出了引理 3，根据该引理可以使拓扑发现算法在地址转发表不完整的情况下得以实现，但是该算法的一个弊端就是不能够发现不支持SNMP协议的网络设备，如HUB。本文给出了改进算法，该算法克服这些缺点，提出了更简单的算法实现。

下面给出了算法所涉及的术语，这些术语的定义在文献[5]中全部给出：S_{ij}(第i台交换机的第j个端口)，A_{ij}(MAC地址集合)，叶端口，标志结点，上行端口，下行端口。

如果一台交换机的所有下行端口都为叶端口，则称该交换机为叶交换机。

本文将除叶交换机、根交换机结点以外的其它交换机统称为中间交换机结点。

引理 1 若路由器R与交换机S_i的S_{ij}端口直接相连，当且仅当S_{ij}是叶端口，A_{ij}仅包含路由器R的MAC地址。

定义 端口S_{ij}的地址转发表A_{ij}是完整的，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地址必然出现在A_{ij}中。同一个子网中的交换机进行通信时，遵循最小“生成树”协议。

如图 1 所示，主机 A 依次向子网中的交换机和与该子网相连的路由器的端口发送 ICMP 响应请求包，由文献[5]标志结点的定义可知，R 为标志结点。由以太网交换机的学习特性可知，数据包从树的根结点经中间结点到达各个叶结点，R 的 MAC 地址就会出现在数据包所流经的所有端口的地址转发表中。由文献[5]对上行端口的定义可知，包含 R 的 MAC 地址的这些端口就是上行端口。各个交换机结点接到请求后，均向 A 发出 ICMP 响应应答包，数据包的流向是从网络中各结点经中间结点到达根结点。此时各结点均已将它的子结点的 MAC 地址记录在相应端口的地址转发表中。完成上述过程后，经过分析发现，每个下行端口所记录的地址转发表的信息都是完整的，但是每个上行端口的地址转发表信息不能保证其完整性。

引理 2 若交换机S_i与S_k满足[5]

$$A_{ij}=\bigcup_{n=0}^{n=N}A_{kn}\bigcup\{S_k\}$$

其中，n =1,2,...,N，N为交换机S_k的端口计数，且n不等于上行端口集合中端口的编号。则S_{ij}与S_k的上行端口S_{kl}直接相连。

根据引理 2，若S_k为叶交换机时，叶交换机没有子树，它的下行端口集合为空。判断端口互连的条件变为：若交换机S_i与S_k(S_k为叶交换机)满足$A_{ij}=\{S_k\}$，则S_{ij}与S_k的上行端口S_{kl}直接相连(如在图 1 中，S_{32}与S_{61}相连的充要条件是$A_{32}=\{S_6$的MAC地址})。为了进一步应用该条件，文献[5]中的算法规定每找出一子结点的连接后，就将这个子结点从树中删去，新生成的树是原来生成树的子树。(如图 1 中的生成树经过一次化简，删去S_6变成图 2，原来的中间结点S_3变成了叶结点)。

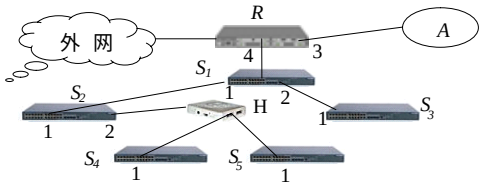

图 2 从图 1 中删去S_6

4 算法改进

依据上面给出的算法思想，如图 2 中$A_{22}=\{S_4, S_5\}$，这时根据引理 2 就不能得到正确的连接关系。本文给出了通过实践而得出的定理及该定理的证明，依据该定理可以通过一种非常简单的方式进行物理拓扑发现，而且可以发现那些不支持SNMP协议的网络设备(交换机、HUB)文中称为“哑”设备，如图 2 中与交换机S_2相连的HUB，从而得到完整的网络拓扑图。

在算法的实现过程中，使用了两个队列，一个为待检测队列，队列中的成员为当前生成树中的中间交换机结点，另一队列为叶子交换机队列，队列中成员为当前生成树中叶交换机结点。算法的过程就是寻找叶子交换机队列与待检测队列中元素间的连接关系。值得注意的是，首先初始化待检测队列为通过发送 ICMP 包而得到的所有活动的交换机构成的队列。

4.1 定理的提出

定理 在下行端口地址转发表完整的情况下，如果中间交换机结点S_i的下行地址转发表中所含记录是该子网中所有中间结点的下行地址转发表中含记录最少的中间结点之一(包括所含记录数相同的情况)，那么该结点下行端口地址转发表中的记录所对应的结点均为叶子结点。如果该中间交换机结点S_i某下行端口地址转发表$A_{ij}=\{S_k\}$(根据引理 2)可以判断出交换机端口S_{ij}和交换机S_k的上行端口直接相连，如果$A_{ij}=\{S_j,\ldots,S_k\}$则说明该端口通过HUB与$S_j,\ldots,S_k$相连。

证明 假设交换机S_i是下行端口地址转发表含记录最少的中间结点，A_{ij}为该交换机任一下行端口的地址转发表。如果A_{ij}中包含网络结点S_t且S_t不是叶子结点，那么一定存在一个结点S_p通过S_t的某下行端口S_{tk}与S_i相连，这样可知，S_t的下行地址转发表所含记录是S_i下行地址转发表所含记录的子集，推出S_i的下行端口的地址转发表不是含记录最少的结点这与已知矛盾，证明完毕。

4.2 改进后的算法描述

具体算法如下(第 6 步~第 10 步为改进后的算法)：

第 1 步 通过异步发送 ICMP 包来判断子网内所有活动的网络设备(交换机和路由器)，通过这一步骤可保证每个交换机所有下行端口的地址转发表信息都是完整的，同时也将所有这些活动交换机初始化为待测交换机队列(除了初始化阶段以外，其它步骤中所提到的待测交换机队列则是由中间交换机结点所构成的)。

第 2 步 从活动的设备当中获得 sysServices 的值，根据第 1 节给出的判断方式区别出交换机和路由器。

第 3 步 得到标志结点的 MAC 地址(通过该标志结点来区别出上下行端口，如第 4 步)。这一步中分为 2 种情况：(1)管理站和待测子网处于同一子网时，标志结点就是该工作站 MAC 地址；(2)管理站

（下转第 115 页）