

基于并行冗余协议的网络拓扑快速发现方法

李 周, 刘 健, 程子敬

(航天恒星科技有限公司, 北京 100086)

摘 要: 针对冗余以太网网络拓扑发现的实时性问题, 提出一种基于并行冗余网络协议的网络拓扑结构快速发现方法, 包括在网络管理端根据网络需求, 发送拓扑发现请求报文, 端系统设备在接收到请求之后周期性地向管理端发送拓扑发现报文, 管理端根据接收到的报文信息进行运算, 从而得到网络的拓扑结构。通过 OMNET++网络仿真软件进行仿真验证, 结果表明, 该方法的网络拓扑构建时间维持在60 ms, 并且实现了与简单网络管理协议的兼容。

关键词: 冗余以太网; 并行冗余协议; 拓扑发现; 端系统; 管理端

Fast Network Topology Discovery Method Based on Parallel Redundancy Protocol

LI Zhou, LIU Jian, CHENG Zi-jing

(Space Star Technology Co., Ltd., Beijing 100086, China)

【Abstract】 As for the real-time problem exists in the redundancy network, this paper presents a fast network topology discovery method based on Parallel Redundancy Protocol (PRP). According to the network requirements, a request packet is sent by the network management system, the agent is sending the topology discovery packet when it receives the packet. Based on the results processed by the network management system, people can obtain the network topology. The method is demonstrated by the platform of OMNET++, results show that time of the network topology discovery is about 60 ms, and this method is compatible with Simple Network Management Protocol (SNMP).

【Key words】 redundancy Ethernet; Parallel Redundancy Protocol (PRP); topology discovery; end system; management end

DOI: 10.3969/j.issn.1000-3428.2012.24.024

1 概述

随着计算机网络技术、空间技术的飞速发展, 两者的交叉领域成为研究的热点, 如何在空间飞行器内部搭建高可用的网络应用平台成为空间计算机网络研究的关键技术。以太网高可用性自动化网络标准 IEC-62439 提出“冗余到网络”和“冗余至节点”2类以太网冗余方案。冗余到网络的典型代表有 STP (Spanning Tree Protocol)、RSTP (Rapid Spanning Tree Protocol) 和 MSTP (Multi Spanning Tree Protocol)^[1]。该方案实现成本低, 但是存在一定的故障恢复时间, 且无法解决因节点网卡故障引起的通信中断^[2]。冗余至节点则为网络节点和2个独立的以太网进行连接, 节点同时在2个网络上发送数据, 从而能消除网络的故障恢复时间。典型代表为基于并行冗余协议 (Parallel Redundancy Protocol, PRP) 协议^[3]的冗余网络。

基于 PRP 的并行冗余网络可以大大提高网络的可靠性, 但同时也提高了对其进行有效维护和管理难度, 而实时正确地获取网络的拓扑结构是人们管理、维护网络的重要基础。

网络本身具有动态性, 单纯依靠人工维护的方式变得

几乎不可能。因此, 网络拓扑快速发现技术的研究是当前网络测量和管理领域的一个重要研究课题。

本文通过对并行冗余协议 (Parallel Redundancy Protocol, PRP) 的分析, 提出一种网络拓扑快速发现方法。通过采用“一次请求, 多次应答”的模式, 实现了网络拓扑结构的快速发现。

2 PRP 协议介绍

PRP 在终端节点实现冗余功能, 而不是在交换机上实现。这是与 RSTP 协议或媒体冗余协议 (Media Redundancy Protocol, MRP) 的主要差别。终端节点连接到2个独立的局域网, 使得2个网络本身的故障相互独立。

图1的PRP网络模型图显示了2个交换网络, 可以为任何拓扑, 例如树形、环形或网状。LAN_A 和 LAN_B 2个局域网在 MAC 层的协议相同, 但可以在性能和拓扑结构上不同, 传输延迟也可能有所不同。它们之间没有直接连接, 被假定为故障无关。在一些应用中, 只有关键节点需要连到2个网络上。为了满足特定的要求, PRP 定义了2种不同类型的终端节点:

作者简介: 李 周(1983—), 男, 硕士研究生, 主研方向: 网络安全; 刘 健、程子敬, 研究员

收稿日期: 2012-02-17 **修回日期:** 2012-04-17 **E-mail:** 513141785@qq.com

(1)双连接节点(DANP)。连接到 2 个局域网。借助链路冗余实体(Link Redundancy Entity, LRE)模块实现数据包的去冗余。

(2)非关键节点(SAN)。可以只连接到一个 LAN 上, 因此, 被称为单连接节点。可以与同一个局域网上的其他节点通信。

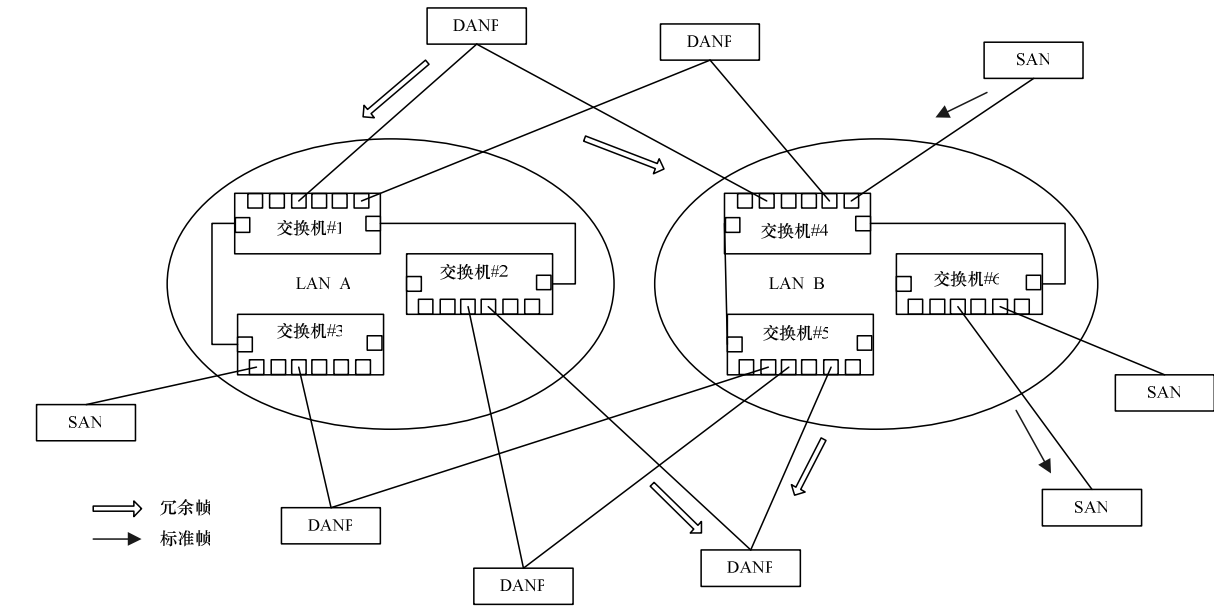


图 1 PRP 网络模型

PRP 协议在网络关键节点上进行实现。图 2 为数据包在 LRE 模块中的处理流程模型。

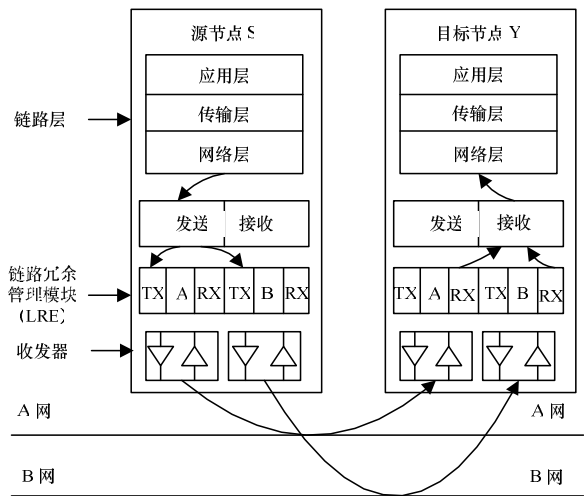


图 2 LRE 数据包处理模型

当上层协议发送一个帧, LRE 复制该帧, 几乎在同一时间从 2 个端口发送出去。2 帧经不同的延迟通过 2 个局域网。最理想的情形是它们在一个小的延迟时间内到达目的地。接收节点的 LRE 向上层转发收到的第 1 个帧, 丢弃重复帧。

LRE 有 2 个任务: 重复帧的处理和冗余的管理, 对于上层来说具有相同的接口。为了监督冗余, LRE 为每个发送的帧增加 32 位的冗余控制标志(Redundancy Control Trail, RCT), 并在接收端去掉此标志。

DANP 的 2 个端口具有相同的 MAC 地址, 并且只有一个 IP 地址。这使得冗余对上层透明, 使 DANP 与 SAN 的地址解析协议(ARP)工作完全相同。

3 拓扑快速发现方法设计

首先给出 6 个相关假设:

(1)端系统和交换机的 OID 表示规则见表 1。

表 1 OID 表示规则

设备名称	OID 表示方法	备注
网管设备	0 或者 1	一级
交换机	00 ~11	二级
交换机	000~111	三级
交换机	0000~1111	四级
交换机	00000~11111	五级
...
端设备	特殊字符 暂定为(A~Z)	端系统

(2)假设存在一组序列 $s_n = \{n_1, n_2, \dots, n_k\}$, 其中, n 为报文序列编号; k 为设备级数; $s_n[i]$ 为序列中第 i 个元素。

(3)拓扑报文的发送者为端设备。

(4)网络中配置一台网络管理设备。

(5)中继设备(交换机)支持路由算法。

(6)冗余报文处理模式: 丢弃冗余报文。

3.1 过程描述

拓扑发现过程包括 6 个步骤:

(1)端系统设备在收到管理端发送的拓扑请求报文之后, 立即周期性地发送拓扑发现报文(自定义发送周期)。

(2)中继网络设备在收到拓扑报文之后, 在报文中相应的字段追加本身的 OID 号, 然后进行转发。

(3)网管设备会不断收到拓扑报文 $s_1 = \{n_1, n_2, \dots, n_k\}$ 。

对于 $s_1 = \{n_1, n_2, \dots, n_j\}$ 和 $s_2 = \{m_1, m_2, \dots, m_j\}$ (因为仅有一台网络管理设备进行报文的接收, 所以可以得到 $n_1 = m_1$), 令 $\Phi = s_1 \cap s_2$ 。

(4)若 $\Phi = n_1 (n_1 = m_1)$, 那么, n_2 和 m_2 以 n_1 为根节点建立拓扑结构, n_3 以 n_2 为父节点, m_3 以 m_2 为父节点建立拓扑结构, 以此类推, 直到每个序列的最后一个 OID 值。

(5)若 $\Phi = \{n_1, n_2, \dots, n_j\} (j > 1) (n_1 = m_1, n_2 = m_2, \dots, n_j = m_j)$, 则说明 s_1 和 s_2 共同经过了 OID 号为 n_1, n_2, \dots, n_j 的中间设备, 此时, $\phi[1]$ 为 $\phi[2]$ 的父节点, $\phi[2]$ 为 $\phi[3]$ 的父节点, 直到 $\phi[j]$, 然后 $s_1[j+1]$ 和 $s_2[j+1]$ 分别以 $\phi[j]$ 为父节点, $s_1[j+2]$ 以 $s_1[j+1]$ 为父节点, $s_2[j+2]$ 以 $s_2[j+1]$ 为父节点, 直到各个序列的最后一个 OID 值, 建立树形拓扑结构。

(6)对于接收到的其他序列 s_k 和 s_{k-1} , 重复执行步骤(4)或步骤(5)。

3.2 程序执行流程

程序执行流程如图 3 所示。

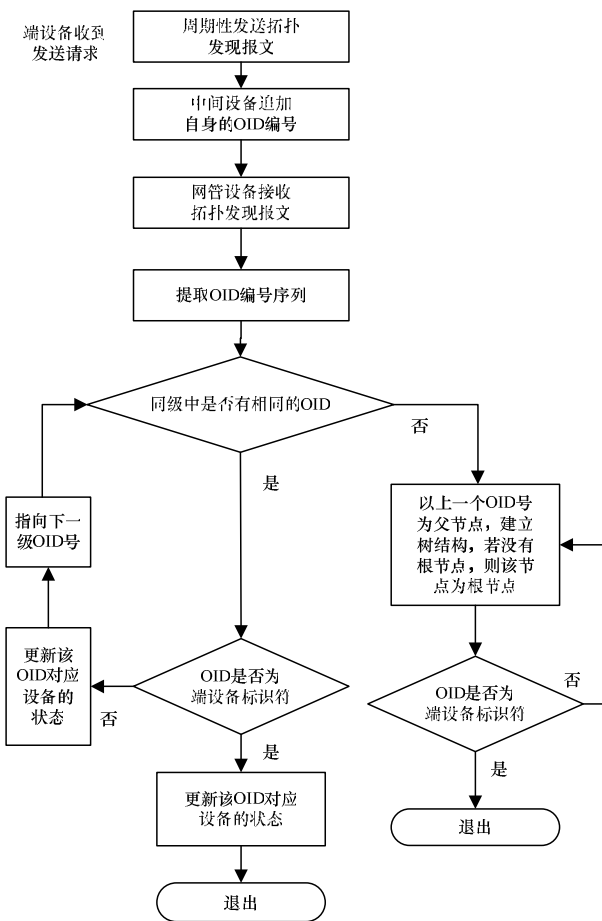


图3 拓扑快速发现程序执行流程

3.3 帧格式定义

拓扑发现报文由每个 DANP 设备的 LRE 模块周期性地产生, 通过 UDP 协议进行传输, 因为网络拓扑发现属于网络管理的内容, 所以设计中考虑到与标准网络管理协议的兼容性, 帧格式设计为 SNMP 协议中类似于 Trap^[4-6]报文的格式, 如图 4 所示。如果该设备为 DANP 设备, 那么在 CRC 字段之前需要插入 RCT 字段, 其中:

A/B: A 网发来的数据或 B 网发来的数据;

OID: 网络设备编号;

Time-stamp: 报文产生的时间;

Extends: 扩展项。

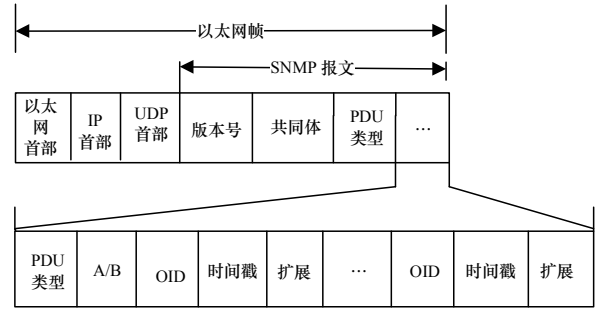


图4 拓扑发现报文字段定义

一个拓扑报文可以包含多个设备的信息(OID, Time-stamp, extends), 只要帧长小于以太网的最大帧长。

拓扑请求报文格式如图 5 所示。

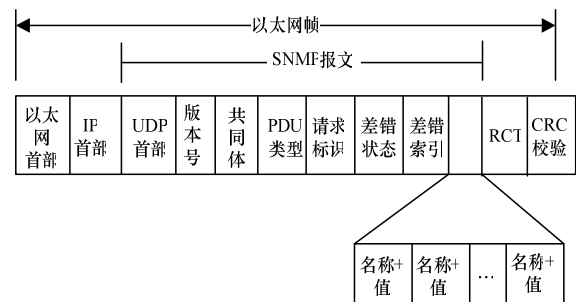


图5 拓扑请求报文字段定义

PDU 类型为 6, 表示类型为拓扑报文, CRC 校验字段前加入 RCT, 以在接收端执行冗余算法(其他字段和标准的 SNMP 协议的规定相同)。

4 基于 OMNET++的网络仿真

本文主要对 DANP 设备 LRE 实体进行建模(图 6), 并借助于标准的交换机模块搭建仿真平台, 对所设计的方法进行验证。

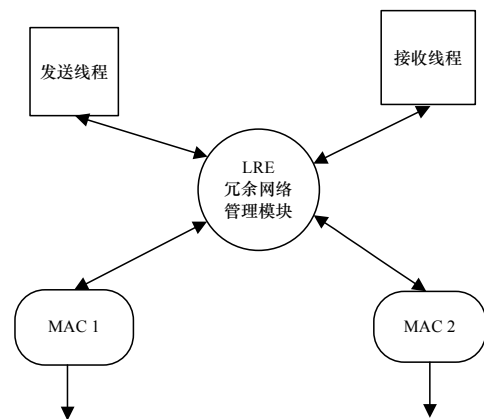


图6 LRE 仿真模型

4.1 基于 PRP 的网元建模

本文对 DANP 设备进行建模, SAN 和交换机采用 OMNET++提供的标准模块。DANP 模块的组件包括数据发送与接收模块、LRE 模块、MAC 模块。

4.2 拟接入网络的设备信息列表

设备初始化列表如表 2 所示。

表 2 设备初始化列表

设备名称	OID	备注
网管服务器(NMS)	0	第 1 级
交换机(etherswitch2)	00	第 2 级
交换机(etherswitch5)	01	第 2 级
交换机(etherswitch1)	000	第 3 级
交换机(etherswitch3)	001	第 3 级
交换机(etherswitch4)	010	第 3 级
交换机(etherswitch6)	011	第 3 级
端设备(danp)	A	第 4 级
端设备(danp1)	B	第 4 级
端设备(danp2)	C	第 4 级
端设备(danp3)	D	第 4 级
端设备(danp4)	E	第 3 级
端设备(danp5)	F	第 4 级
端设备(SAN)	G	第 4 级
端设备(danp7)	H	第 3 级
端设备(danp8)	I	第 3 级

4.3 报文格式定义及发送

拓扑请求报文定义如下:

(1)管理端发送的请求报文结构体

```
struct messageStruct {  
    unsigned char buffer[1450]; //用于存放从 161 端口接收的报文  
    int  len; //指示收到报文的真实长度  
    int  index; //当前字节的索引  
};  
//index 在程序实现中用于指示当前处理到的报文的位置  
switch (msg[index]) {
```

```
case  SEQUENCE;  
case  GET_REQUEST;  
case  GET_NEXT_REQUEST;  
case  SET_REQUEST;  
  
case  GET_TOPOLOGY; //拓扑请求  
case  NO_TOPOLOGY; //停止发送拓扑发现报文  
};  
//构造好的输出报文通过协议栈接口发送  
sendto(snmefd,  response.buffer,  response.index,  0,  (struct  
sockaddr *)&from, fromlen);
```

(2)拓扑发现报文的报文结构体

```
struct messageStruct {  
    unsigned char buffer[1450]; //162 端口接收  
    int  len; //指示收到报文的真实长度  
    int  index; //当前字节的索引  
    int  LANindex; //表示是属于不同局域网  
};  
//构造好的输出报文通过协议栈接口发送  
sendto(snmefd,  response.buffer,  response.index,  0,  (struct  
sockaddr *)&from, fromlen);
```

(3)报文的接收

```
//接收来自 socket 套接字:  
request.len= recvfrom(snmefd, &request.buffer[0], 1024, 0, &from,  
&fromlen);
```

然后进行合法性检验, 检验通过后提交给应用程序, 根据拓扑发现方法进行运算之后即可得到网络拓扑结构, 见图 7 和图 8。

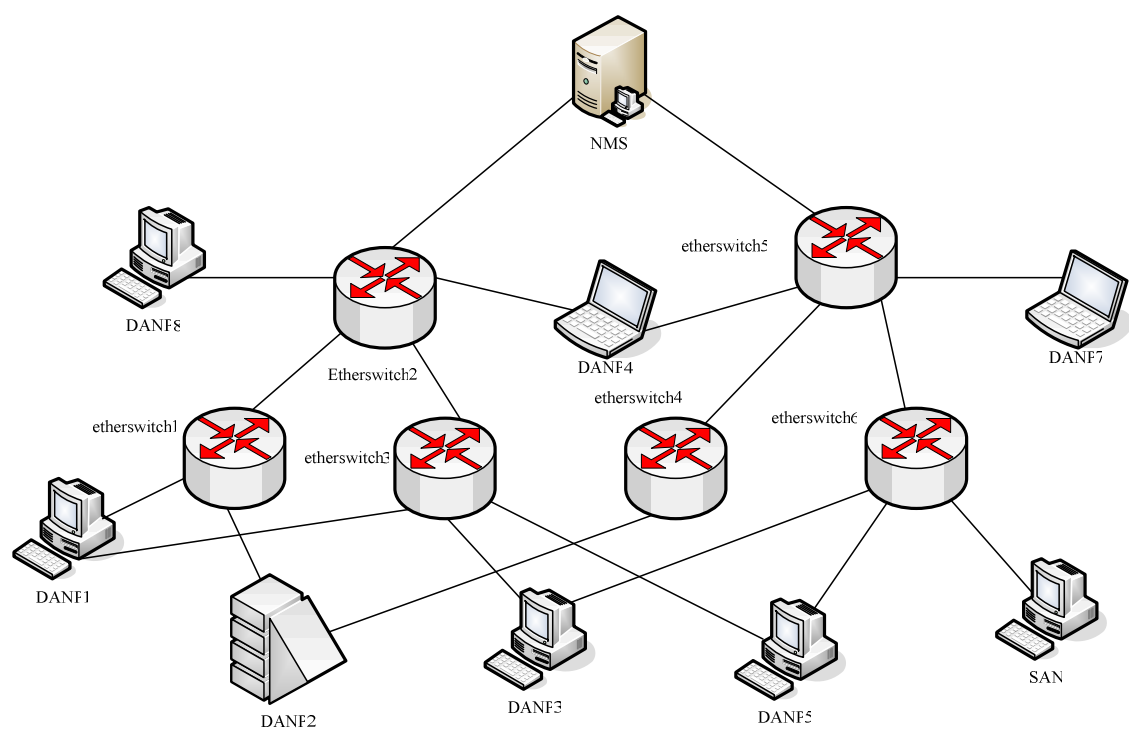


图 7 拓扑发现结果 1

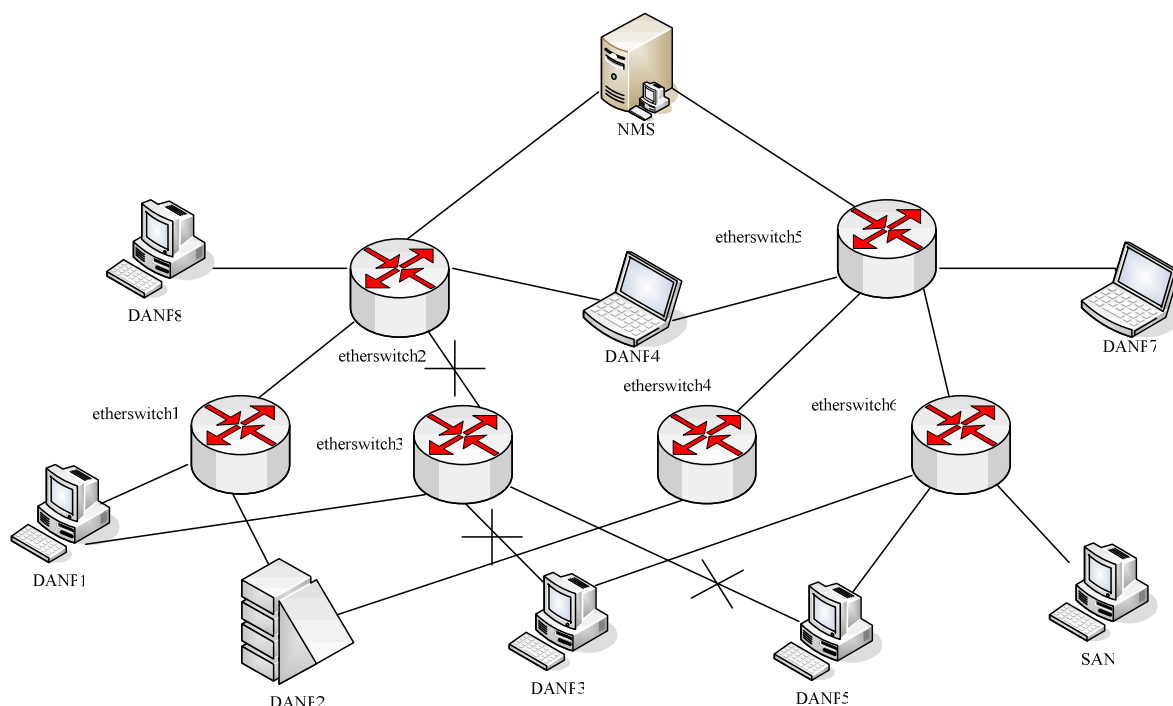


图 8 拓扑发现结果 2

4.4 仿真结果

15 台设备全部被发现及这些设备的逻辑连接需要约 60 ms。采用常规的基于 SNMP 的网络拓扑发现则至少在秒级(2 台主机和 1 台商用交换机搭建的测试环境,其中,商用交换机内置支持 SNMP 协议的 Agent),并且随着网络规模的增大,拓扑发现的时间会增加。

设置 etherSwitch2 到 ethersSwitch3 的交换机内部链路断开,那么图 7 到图 8 的转换时间约 90 ms。

单点的故障并没有影响整个网络的正常运行,通过拓扑发现能够及时知道备份网的运行情况,故障范围定为在 ethersSwitch3、ethersSwitch3 与 ethersSwitch2 之间的连接及相应的端口。

4.5 结果分析

发现效率: 冗余网络拓扑结构快速发现方法能够快速地发现 A 网和 B 网内的所有交换机及 DANP 设备及其复杂的连接关系。虽然 G 设备为 SAN 设备,但是由于其内部具有 Agent(SNMP 网管代理),与该方法是兼容的,因此也能够被发现。

实时性分析: 通常的拓扑发现方法都是基于“应-答”模式,如基于 SNMP 的拓扑发现算法^[7-9]、基于交换机地址转发表的方法^[10],而本文方法通过定时向管理端发送拓扑报文,可以减少一半的时间延迟。而且网络规模越大,这种优势与基于 SNMP 协议的拓扑发现方法相比更明显。

网络规模对拓扑构建的影响: 本文方法是终端设备在接到请求之后主动向管理端发送拓扑发现报文,即“一次请求,多次应答”的模式,因此,在小型局域网中,网络拓扑的构成对于网络拓扑的发现时间几乎没有影响,与管理端服务器的处理能力有直接关系,仿真场景选取 15 台

设备具有一定的代表性。

5 结束语

本文提出的基于 PRP 的网络拓扑快速发现方法适用于对实时性要求较高的网络环境,通过采用“一次请求,多次应答”的模式,实现了网络拓扑结构的快速发现。该模式还可以扩展到其他网络管理的功能中,提高网络管理的实时性。

参考文献

- [1] Huynh M, Goose S, Mohapatra P. Resilience Technologies in Ethernet[J]. Computer Networks, 2010, 54(1): 57-78.
- [2] 陈 琼. 工业以太网网络拓扑发现技术的研究[D]. 成都: 电子科技大学, 2006.
- [3] IEC. IEC-62439-3-2010 Industrial Communication Networks——High Availability Automation Networks[S]. 2010.
- [4] Case J, Fedor M. A Simple Network Management Protocol (SNMP)[S]. RFC 1902, 1996.
- [5] Introduction to Simple Network Management Protocol[EB/OL]. (2011-11-11). <http://www.windowsnetworking.com>.
- [6] What Is Simple Network Management Protocol[EB/OL]. (2011-10-22). <http://www.businessdictionary.com/>.
- [7] 王志刚, 王汝传, 王传棣, 等. 网络拓扑发现算法的研究[J]. 通信学报, 2004, 25(8): 36-43.
- [8] Son M H, Joo B S, Kim B C, et al. Physical Topology Discovery for Metro Ethernet Network[J]. ETRI Journal, 2005, 27(4): 355-366.
- [9] 程春玲, 崔国亮, 隋宗见. 基于广播 SNMP 的网络管理并行轮询算法[J]. 计算机应用与研究, 2010, 27(12): 2134-2137.
- [10] 陈 松, 王 珊, 周明天. 基于链路层的拓扑算法研究[J]. 电子学报, 2009, 37(10): 2338-2342.

编辑 张正兴