文章编号:1001 - 9081(2005)04 - 0891 - 03

一种改进的网络拓扑发现方法

邱建林[1],何 鹏[1,2]
(1. 南通大学 计算机科学与技术学院,江苏 南通 226007;
2. 苏州大学 计算机科学与技术学院,江苏 苏州 215006)
(qiu. jl@ ntit. edu. cn)

摘 要:在对基于 ICMP 的网络拓扑发现、基于 ARP 的网络拓扑发现和利用 SNMP 访问 MIB(管理信息库)路由表的网络拓扑发现三种方法的分析研究基础上,提出了一个经过改进的网络拓扑发现方法,此方法能够准确、完整、高效地发现网络主干拓扑和子网内的设备,并详细描述了网络拓扑发现的数据结构和算法。

关键词:拓扑发现;简单网络管理协议;管理信息库;ICMP,地址解析协议
中图分类号: TP393.02 **文献标识码**:A

Ameliorate algorithm for network topology discovery

QIU Jian-lin[1], HE Peng[1,2]
(1. *College of Computer Science and Technology, Nantong University, Nantong Jiangsu* 226007, *China*;
2. *College of Computer Science and Technology, Suzhou University, Suzhou Jiangsu* 215006, *China*)

Abstract: Three kinds of centralized automatic discovery algorithms for network topology were introduced, which include building network toplogy base on ICMP or ARP protocol and utilizing SNMP protocol to visit in MIB(management information base) to construct network toplogy. Then, a kind of better network toplogy algorithm was proposed based on the three algorithms which can discover network topology accurately, unabridged and efficiently. And the data structure and steps of this algorithm was described detailedly.

Key words: toplogy discovery; SNMP; MIB; ICMP; ARP

网络的拓扑结构发现是网络管理的基础,对监视整个网络,获取完整的网络信息,保证网络高效、稳定地运行非常重要。

本文对现今流行的几种网络拓扑发现技术进行了分析,对这些技术的优缺点和适用环境进行了一些探讨,介绍了一种基于 ICMP、ARP、SNMP 改进的网络拓扑发现方法,并给出了实现模型。

1 基于 ICMP 路由表的拓扑发现方法

ICMP[1] 是提取路由信息的重要工具,人们常利用ICMP echo reply 消息来检测网络设备的活动状态和可达性。利用 ICMP time exceed 和 port unreachable 消息,以及 IP 协议中 TTL 字段可以发现给定主机的路由信息。如果对一个网段内所有可能的 IP 地址依次执行"Ping"操作,根据应答就可以发现该网段内所有当前活动的设备。然后对"Ping"通的 IP 地址逐一执行"Traceroute"操作,记录每一次操作的结果,根据前面的操作结果,再分析所得到的信息,从而得到整个网络拓扑的情况。

基于 TCP/IP 协议的网络设备几乎都支持所有的 ICMP 协议,该方法实现起来较容易。在发现网络拓扑的同时,检测

收稿日期:2004 - 09 - 06 **基金项目**:江苏省高校自然科学基金资助项目(03KJB520103);江苏省现代教育技术资助项目(2004 - METR - 27);南通市科委资助项目(C3018、C3024)

作者简介:邱建林(1965 -),男,江苏南通人,副教授,主要研究方向:算法分析与设计、网络安全、逻辑综合 VLSI 的 CAD; 何鹏(1980 -),男,江苏南通人,硕士研究生,主要研究方向:计算机网络安全.

但是,RM 可调度判定算法有自身的缺点,在一些特定现场实时系统中该调度算法并不适用,进而我们提出一种新的硬实时调度算法(HRTCS)方案,并对该算法同 RM 进行了严格对比测试。经实验说明该方案系统开销较小、可实施性好,易于移置到其他硬实时通信系统中。

参考文献:

[1] TINDELL KW, BURNS A, WELLINGS AJ. Allocating hard real-time tasks: an NP-hard problem made easy[J]. Real-Time Systems, 1992, 4(2): 145 - 165.

[2] SHA JPL, DING Y. The Rate - Monotonic Scheduling Algorithm: Exa Characterization and Average Case Behavior[A]. Proceedings of IEEE Real-Time System[C]. 1989. 166 - 171.

[3] BARUAH SK, MOK AK, ROSIER LE. Preemptively Scheduling Hard-Real-Time Sporadic tasks on One Processor[A]. Proceedings of IEEE Real-Time System Symposium[C]. Orlando, Fla, 1990. 182 - 1 - 90.

[4] BAKER TP. A Stack-Based Resource Allocation Policy for Real-time Processors[A]. Proceedings of IEEE Real-Time System Symposium [C]. 1990. 191 - 200.

[5] KRISHNA CM, SHIN KG. Real-Time System[M]. 北京: 清华大学出版社, 2001.

[6] Fixed Priority Scheduling of Periodic Task Sets with Arbitrary Deadlines[A]. Proceedings of IEEE Real-Time Systems Symposium[C]. 1990. 201 - 210.

[7] BURNS A. Scheduling Hard Real - Time Systems: S Review [J]. IEEE, 1991, 112 - 119.

了网络设备的活动状态,对可能因为依次 Ping 各个 IP 地址而带来的管理工作站和网络负担增大的问题,可采用广播方式或多线程方式来减小这一开销。该方法存在的不足:1)"Ping"操作可以较容易地发现活动的网络设备,但是不能确定 ICMP 包从源 IP 节点到目地 IP 节点所经过的路由设备。在使用"Traceroute"操作后可以知道中间经过了哪些路由设备,但是路由设备之间的连接情况却很难分析;2)发现方法本身也具有一定的盲目性,对于给定的 IP 地址,即使 Ping 通了,也很难判断各个 IP 所在子网的情况;3)由于一些防火墙禁止通过 ICMP 包,该方法无法检测存在于这些防火墙后面的设备。因此,基于 ICMP 的拓扑发现方法可以快速发现网络设备,但构建网络拓扑设备关系图比较困难。所以在实际应用中,主要用此方法来发现子网中的网络设备。

2 基于 ARP 的拓扑发现方法

以太网接口的网络设备都支持 ARP[2](地址解析协议),并在本机维护一张 ARP 表,用于 IP 地址与以太网地址间的地址解析和转换。由于在同一以太网段内所有的活动主机的地址信息一般都在 ARP 表中,因此可以利用 ARP 表来进行拓扑发现。根据任何一台路由设备或交换机的 ARP 表,可以发现与其各个以太端口连接的以太局域网中的所有网络设备,再根据其他信息判别网络中的路由设备和交换机,并继续根据其 ARP 表进行发现,可以得出整个以太网的拓扑结构。

ARP 表中的网络设备地址都是最近活动过的有效 IP,而且几乎没有冗余信息。所以采用此方法的发现效率很高。

该方法存在的不足:当网络过大时,ARP 表中的记录可能无法包括网络中实际存在的所有网络设备。所以此法用于局域网内设备的发现比较合适。

3 基于 SNMP 的拓扑发现方法

SNMP[3]是一种基于 IP 协议的协议,利用 SNMP 可以从网络设备中提取用来支持网络管理的 MIB[4](管理信息库)信息。现在用做网关的网络设备都支持 SNMP 代理,网络拓扑信息主要包含在 MIB 中,通过对 MIB 中拓扑信息的获取,就可以分析出网络的拓扑连接情况。

路由信息表 ipRouteTable 定义了该设备中的路由信息(ipRouteDest, ipRouteIfIndex, IpRouteNextHop, IpRouteType, …)。ipRouteDest 是一个索引,它记录了以该设备为根节点可能到达的各个目标地址或地址范围;ipRouteIfIndex 记录了与 ipRouteDest 对应的端口索引号;IpRouteNextHop 记录了本网关下一跳的地址;IpRouteType 记录了目的子网与该网的关系。路由表中的下一跳地址 IpRouteNextHop 所标识的必然是具有路由功能的网络节点,所以从网络管理工作站的默认网关开始,通过读取路由设备的路由表下一跳信息,由内向外发现网络中的所有具有路由功能的网络节点及其连接关系。而从 IpRouteType 的值就可以知道该网关连接哪些子网。同时通过路由表的 ipRouteIfIndex 就可以发现端口,通过由接口表得到端口的类型就可以了解子网类型。由上所述就可以得到网络的主干拓扑关系。

该拓扑发现方法完全基于标准的 SNMP 来实现,发现过程的算法简单,系统和网络开销小,所以效率较高。该方法存在的不足:1)无法发现网络中不支持 SNMP 协议或没有安装 SNMP 代理的网络设备;2)路由表中包含了大量的冗余信息;3)一个路由往往对应多个 IP 地址(每个端口对应一个 IP),而路由表是以唯一的 IP 地址作为记录因素,这样就存在多目路由的问题,不利于直观的反映网络拓扑的连接情况。所以此法适合于发现网络中的主干拓扑,反映网络的整体状况。

4 改进的网络拓扑发现方法

通过对基于 ICMP、ARP、SNMP 协议网络拓扑发现方法的分析和比较,发现它们具有各自的优点和不足,如果将它们结合起来使用,可以达到更好的效果。本文采用分级发现的策略,将网络拓扑发现分成两级进行:第一级用于网络拓扑发现中的路由(网关)设备和子网;第二级用于拓扑发现子网内的主机设备。为了更直观的表示设备,还应获取设备所拥有服务及其他信息,才能更好地体现网络的层次。

4.1 一级拓扑发现

一级拓扑发现主要采用基于 SNMP 协议访问路由设备路由表的方法[5]。这种方法在前面已经介绍过,从网管工作站所在子网的缺省路由开始,采用广度优先遍历无向图的算法对整个网络系统的路由设备进行遍历,并综合分析路由信息表,就可以发现网络系统的主干拓扑结构。但此方法存在不足,即将多重身份的路由设备(多个 IP)作为多个路由设备处理。

对此不足的解决方法:地址表 ipAddrTable 中定义了 ipAdEntAddr 变量,用来表示接口的 IP 地址。显然,只要遍历该地址表就可以获得该路由设备的所有 IP。根据 RFC1519,亚太地区分配到的 C 类网地址为:202.0.0.0 ~ 203.255.255.255。所以国内合法的 C 类网 IP 应在 202.0.0.0 以上。而实际上部分不合法的子网 IP 往往选择 192.168.xxx..xxx。故笔者认为选取 ipAddTable 中的最大 IP 来唯一代表路由设备比较合理,取得的 IP 地址是合法的可能性更大。

一级拓扑的遍历查询,遵循下面的原则[6]:1)如果路由的类型是不直接相连的(ipRouteType = 4),被查询路由表的网关是与路由表中的 ipRouteNextHop 相邻的;2)如果路由的类型是直接相连的(ipRouteType = 3),ipRouteDest 所表示的子网是与查询的网关是相邻的;3)对遍历查询设置发现时间,用来控制何时结束发现。

4.1.1 一级拓扑发现的关键数据结构

实现中关键的数据结构有:

```
(1) ipaddressCurrentRouter;
//当前路由:与当前网关相对应的路由设备的唯一标识,即当前
//网关地址表中最大的 ipAdEntAddr 值。
(2) ipaddressCurrentGateway;
//当前网关:当前正在访问的路由设备的路由端口
(3) structRouterQueueItem
    {
      //路由队列:用于存放发现到的所有路由设备的标识
      ipaddressRouterAddr;                    //路由地址
      structRouterQueueItem * next;  //队列的下一个元素指针
    }
(4) struct VisitedGatewayQueueItem
    {
      ipaddressRouterAddr;                    //网关地址
      structRouterQueueItem * next;  //队列的下一个元素指针
    } VisitedGatewayQueueItem, UnVisitedGatewayQueueItem;
          //VisitedGatewayQueueItem 已访问网关地址队列
```

```
        //UnVisitedGatewayQueueItem 未访问网关地址队列
(5) structSubnetQueueItem
    {
      //子网队列:用于存放发现到的所有子网
      ipaddress SubnetAddr;                    //子网地址
      ipaddress SubnetMask;                    //子网掩码
      struct SubnetQueueItem * next;
                                  //队列的下一个元素指针
    };
(6) structConnectionQueueItem
    {
      //连接队列:用于记录所有路由与路由,路由与子网的
      //连接情况
      ipaddress From;                  //连接的一方地址
      ipaddress To;                    //连接的另一方地址
      structConnectionQueueItem * next;
                                  //队列的下一个元素指针
    };
```

4.1.2　一级拓扑发现的实现

(1)初始化路由队列、待访问网关队列、已访问网关队列、子网队列和连接队列,设置发现时间;

(2)读取网络管理工作站的本机系统文件/etc/sysconfig/network,取得其默认网关的 IP 地址;

(3)令(2)中取得的 IP 为默认网关,将其加入待访问网关队列;

(4)从待访问网关队列中取出一个节点,作为当前网关并开始记时;

(5)访问当前网关的 ipAddrTable 表,取出 ipAdEntAddr 的所有值,添加到已访问网关队列。将 ipAdEntAddr 的最大值添加到路由队列,并将其置为当前路由;

(6)如果没有超时则执行(6),否则执行(13);

(7)绑定对象 ipRouteNextHop、ipRouteDest、ipRouteMask、ipRouteType,采用 SNMP 协议中的 GetNextRequest 操作,读取当前网关 ipRouteTable 第一行中的这四个值;

(8)分析(6)取回的 ipRouteNextHop,如果已访问网关队列和待访问网关队列中没有该值,则将其添加到待访问网关队列;

(9)如果取回的 ipRouteType = 4,则访问 ipRouteNextHop 的地址表,取出其最大 ipAdEntAddr 值。如果路由队列中没有该值,将它加入。构造一个 ConnectionQueueItem 节点,将该值赋值给 ConnectionQueueItem. To,而当前路由赋值给 ConnectionQueueItem. From;

(10)如果取回的 ipRouteType = 3,并且子网队列中没有该子网,则将它加入子网队列。子网掩码为取回的 ipRouteDest 的值,子网掩码为取回的 ipRouteMask 的值;构造一个 ConnectionQueueItem 节点,将取回的 ipRouteDest 赋给 ConnectionQueueItem. To,而当前路由赋给 ConnectionQueueItem. From;

(11)绑定(7)取回的对象值,再次调用 GetNextRequest,继续取下一行的值,重复(6)~(10),如此循环直到表访问结束。最后,将当前网关不重复的加入已访问网关中队列,并从待访问网关队列中删除;

(12)如果待访问网关队列没有访问完,则取下一节点为当前网关,重复(4)~(11);

(13)结束。

4.2　二级拓扑发现

4.2.1　确定子网内网络设备的状态

子网内主机的当前状态一般使用 ICMP Ping 方法来检测。但如前面提到,对子网内可能的 IP 地址采取依次 Ping 的方法会有一定的盲目性,导致网络负担增大。将基于 ICMP 的拓扑发现方法与基于 ARP 协议的拓扑发现方法相结合,首先访问子网所连接的网关的 MIB 库中的 ipNetToMediaTable 表,利用表中的变量 ipNetToMediaNetAddress 确定子网中实际连接的主机的 IP 地址。然后再 Ping 这些实际的 IP 地址,确定主机状态。这样很大程度上减少 Ping 的次数,提高了算法的效率,减少了网络负担。此外还可以使用异步 Ping 的方法来缩短检测时间。

4.2.2　确定子网内网络设备的类型

网络设备的类型可以通过访问 MIB 库的 ipForwarding 和 sysServices 变量来确定。ipForwarding 变量用于说明网络设备是否具有向前转发 IP 数据包的功能,如具有这项功能,则此值为 1,说明设备为网关;如不具有这项功能,则该值为 2。sysServices 的值表示网络设备可以提供的服务集合,根据它也可以判定该设备是工作在 OSI 的第几层。例如 sysServices = 2,说明工作在数据链路层,为桥接器。由此可见,根据该值可以判断出一个网络设备的服务层次。

4.2.3　确定子网内网络设备的网络接口

对于网络接口,在 MIB 库中使用接口表(ifTable)描述,接口表中的 ifType 字段用来描述接口的类型,使用一个整数表示,其说明如表 1 所示。

表 1　网络接口类型表部分

IfType 值	代表的网络接口
4	Ddn-x.25
6	Ethernet-csma/cd
9	Iso88025-tokenRing
15	fddi

通过访问接口表中 ifType 字段的值,对照网络接口类型表就可以确定网络设备的网络接口。

5　结语

本文提出的改进算法是采用分级搜索的策略,采用基于 SNMP 协议访问路由设备路由表,将基于 ICMP 协议的拓扑发现方法与基于 ARP 协议的拓扑发现方法相结合,确定子网内网络设备的状态、类型、接口。随着网络连接的复杂性和网络协议的多样性,要想对于各种网络的各种设备的发现,应根据实际需要运用多种技术和方法来共同完成。

参考文献:

[1] 王建国. 基于协议集成测试系统的 IPv4 协议一致性测试[J]. 软件学报,2000,11(2):207~212.

[2] CARL MITCHELL S, QUARTERMAN JS. Using ARP to Implement Transparent Subnet Gateway, RFC 1027[S]. 1987.

[3] CASE J. Simple Network Management Protocal(SNMP), RFC 1157 [S]. 1990.

[4] ROSE M. Management information base for network management of TCP/IP-based internets: MIB-II, RFC 1213[S]. 1991.

[5] 李天剑,曾文方,李天翼. 基于 SNMP 网络拓扑自动构造的一种实现[J]. 计算机系统应用,2000,(2):29~31.

[6] 孙德文,高儒振. 基于 SNMP 网络拓扑图的自动构造实现[J]. 上海交通大学学报,1997,31(8):59~64.