**网络拓扑发现算法的分析**

**摘　要**： 介绍了几种常见的网络拓扑发现工具，从负载、速度、准确性及适用范围几个方面对各工具的执行效果进行了对比；分类归纳了常用的网络拓扑发现的方法；分析了利用这些方法实现的七种拓扑发现算法，并针对每种算法详细列出了其优缺点。给出了对网络拓扑发现算法进行评价的标准及其量化的表示形式。  
**关键词**： 拓扑发现 算法 SNMP ICMP

　　近年来，由于计算机网络的迅速发展，有效的网络管理得到了越来越多的重视，而获得最新的网络拓扑结构对于网络管理至关重要。因为网络具有动态特性，并且网络规模日益增大，利用人工维护网络的拓扑图几乎不可能。因此，网络拓扑自动发现技术的研究广泛开展起来。Cornell大学的CNRG研究组、美国南加州大学USC(University of Southern California)的SCAN研究组和Internet数据分析合作组织CAIDA(Cooperative Association for Internet Data Analysis)等都在该领域进行了大量的研究工作，并取得了很多成果。网络拓扑发现涉及到网络体系结构的各个层次，可以利用很多协议设计不同的算法。各种算法的性能差距很大，各有所长。判断一种算法优劣的指标主要是：负载、速度和准确性等，具体说就是一个算法在不过多增加网络负载的前提下，尽量提高收敛速度和准确性。以前的网络拓扑发现算法主要集中精力在发现网络的逻辑拓扑(基于第三层的)，这些算法忽略了第二层的网络设备(交换机或网桥)的连接。即使提供了第二层的网络拓扑发现，也是针对具体的网络设备，不具有通用性。近年来，出现了一些通用的基于第二层的网络拓扑发现算法。

**1 拓扑发现的常用工具  
1.1 ARP协议**  
　　每个支持地址解析协议ARP(Address Resolution Protocol)的网络设备中都维护着一张ARP表，该表中记录了该设备所连接的以太网中网络设备的IP地址和MAC地址的对应关系。根据ARP表的这个特点，可以从一台已知的路由器或交换机的ARP表发现其连接的以太网中其他网络设备，再从这些新发现的设备中区分出路由器和交换机，并继续根据这些设备的ARP表进行网络设备发现。依此类推，就得到了整个网络的拓扑结构。  
**1.2 SNMP协议**　　利用简单网络管理协议SNMP(Simple Network Management Protocol)，一个管理工作站可以远程管理所有支持这种协议的网络设备，包括监视网络状态、修改网络设备配置和接收网络事件警告等。SNMP协议利用管理信息库MIB(Management Information Base)管理网络设备的配置和状态信息，每个支持SNMP的被管理设备都维护了一些MIB库。其中，最常用的是MIBII(RFC-1213)和Bridge-MIB(RFC-1493)。  
**1.3 ICMP协议**　　在网络拓扑发现中主要利用了Internet控制消息协议ICMP(Internet Control Message Protocol)的echo(回应)请求、echo应答和超时这三种报文。拓扑发现中常用的工具Ping就是利用ICMP的echo报文发现网络的连通性，而Traceroute则是利用ICMP超时报文发现网络中两节点间的路径。Mercator算法就是利用启发式规则和限制跳数的Traceroute进行拓扑发现[1]。  
**1.4 DNS Zone Transfer**　　域名服务器DNS(Domain Name Service)中保存了域内的主机名和IP地址的对应关系列表。“zone transfer”命令可以使DNS服务器返回域内所有域名的列表，所以DNS Zone Transfer可被用来发现域内的主机和路由器。  
**1.5 其他拓扑发现工具**　　利用路由信息协议RIP(Routing Information Protocol)、开放最短路径优先协议OSPF(Open Shortest Path First)或边界网关协议BGP(Border Gateway Protocol)可以获得路由设备中存储的路由信息。利用该信息发现新的路由设备并判断设备之间的连接关系。从而发现整个网络的拓扑结构。CNRG提出的拓扑发现算法和CAIDA的Skitter算法都利用了BGP协议。表1列出了上述方法在执行效果(负载、速度、准确性和适用范围)上的对比。

**2 网络拓扑发现常用方法  
2.1 基于第三层的拓扑发现常用方法**　　第三层的网络拓扑发现方法着重于发现路由设备间的逻辑连接关系。它发现的拓扑结构并不表示网络中设备的真正连接关系，而是“IP数据报转发”意义上的连接关系。下文介绍了三种基于第三层的拓扑发现的启发式规则：  
　　规则1：利用广播Ping进行子网猜测。  
　　给定一个IP地址，可利用本规则猜测该地址所属的子网掩码：  
　　for(MskLen=31；MskLen>7；MskLen--) {//构造主机号全为0或全为1的广播地址  
　　BrdcstAddrs=CnstructBrdcstAddr(MskLen)；//ping构造的广播地址  
　　ResultCount=BrdcstPing(BrdcstAddrs)；//若每个广播地址都获得了回复，算法结束，返回当前猜测的子网掩码长度  
　　if(ResultCount>=2)  
　　return MskLen；  
　　}  
　　本规则的基本思想是递减猜测子网掩码的长度，对每个猜测的子网掩码构造广播ping地址，如果主机对构造的广播地址有回应则说明猜测的子网掩码是正确的[2]。  
　　规则2：利用一组地址进行子网猜测。  
　　假设知道地址集A中的IP地址同属于一个子网，可利用本规则猜测这组地址所属的子网地址。  
　　①计算A的按位与BitwiseAND{A}和它的按位或BitwiseOR{A}；  
　　②逐位对比BitwiseAND{A}和BitwiseOR{A}，找出第一个不相同的位；  
　　③将步骤②中找到的位和其后的各位设置为0，猜测子网掩码；  
　　④将步骤③中的结果分别与BitwiseAND{A}进行按位与，算出子网号。  
　　本规则的基本思想是：对比地址集A的按位与和按位或的结果，找出第一个不相同的位，则子网掩码在该位一定为0即该位及其后的各位都只能是主机号，而不可能是子网号。从而判断出子网掩码的长度，而子网地址则可通过BitwiseAND{A}与子网掩码做按位与运算得出[2]。

　　启发式规则2在以太网上的应用示例如图1所示。以图1为例执行本规则,因这组地址的前三个字段相同，故不失一般性，只对第四个字段进行运算：  
　　①BitwiseAND{A}=10000000，BitwiseOR{A}= 10111111；  
　　②BitwiseAND{A}与BitwiseOR{A}在第三位不同；  
　　③子网掩码的最后一个字节一定为ab000000(ab为11，10或00)；  
　　④将ab000000与10000000(BitwiseAND{A})进行按位与,结果形如c0000000(c为1或0)。  
　　规则3：猜测域内的有效地址。  
　　本规则可被用来推测已知的子网地址空间中有效的IP地址。其算法描述如下。  
　　while(AliveIPList Not NULL) {  
　　　　this=getIP(AliveIPList)；//获得this后的N个地址，存入临时地址集  
　　　　GetFollowingIPs(this，N)；//this的最后一个字节为1、65、129或193  
　　　　if(GetLastByte(this)=1 or 65 or 129 or 193)//选取与this具有相同前缀的N个地址，存入临时地址集  
　　　　GetIPsWithSamePrefix(this，N)；  
　　}  
　　N的选择非常重要，若N过大，则能够获得所有的有效地址，但也会包含一些无效地址；若N太小，则发现的大部分地址是有效的，但同时也会遗漏一些有效地址[2]。  
**2.2 基于第二层的拓扑发现常用方法**　　基于第二层的拓扑发现方法着重于发现网络设备端口间的物理连接。基于第二层的拓扑发现常用的2种方法如下。  
　　(1)利用生成树协议  
　　以太网中的交换机和网桥都维护了一个地址转发表，其中为每个端口保存了其转发过的数据帧的源MAC地址，若地址转发表是完备的，则可利用生成树协议推导出如下三条定理，用来判断两个端口间的连接关系：  
　　定理中的Si表示交换机i；Sij表示交换机i的第j个接口；Aij表示与Sij相关的地址转发表中的MAC地址的集合，即Aij中的MAC地址都是从Sij收到的数据帧的原地址；Uijkl 表示Aij∪Akl。  
　　①假设Sij和Skl是不同的接口，如果Aij和Akl的交集不为空，则Sij与Skl没有直连[3]；  
　　②假设t是至少包含两台交换机Sp和Sq的子网，如果Aij和Akl的交集为空并且Uijkl包含且仅包含Sp和Sq中的一个，则Sij与Skl没有直连[3]；  
　　③假设Aij和Akl的交集为空，且Aij和Apt的交集为空，如果Uijkl=Uijpt并且Si和Sk属于同一个子网而Sp属于不同的子网，则Sij与Skl没有直连[3]。  
　　(2)利用端口的流量统计  
　　通过对网络设备的各个端口的流量数据进行统计，并结合其他的第二层网络拓扑发现方法，可判断出端口间的直连关系。

**3 网络拓扑发现的常用算法  
3.1 基于SNMP和Ping的算法**  
　　算法的主要步骤是将探测源模拟成一个网管站与SNMP Agent通信，先取得探测源的默认网关，存入待探测队列ToDoList。依次取出ToDoList中的IP，获得该IP的MIB库中的ipRouteTable中的数据。当ipRouteType= indirect时，可以利用ipRouteNextHop获得路由器-路由器的连接，并将ipRouteDest存入ToDoList；当ipRouteType = direct时，利用ipRouteDest可以获得当前路由器连接的子网。然后Ping子网内的每个IP地址，析取ICMP回应报文的IP地址，确定出子网内活动主机的信息。当ToDoList所有IP均被处理后，生成网络拓扑结构图。  
**3.2 基于广播Ping和DNS Zone Transfer的算法**　　算法的主要步骤是首先利用DNS Zone Transfer获得域内设备的IP地址并存入临时地址集，然后依次从临时地址集中取出地址，进行以下操作，直到临时地址集为空：利用Ping判断该地址是否有效，若有效则将该地址存入有效地址集，并且利用广播Ping猜测该地址所在的子网地址。Ping该子网的广播地址，将有回复的IP归入该子网，并且加入到临时地址集。  
**3.3 基于Traceroute和DNS Zone Transfer的算法**　　算法的主要步骤是首先利用DNS Zone Transfer获得域内设备的IP地址并存入临时地址集，然后依次从临时地址集中取出地址存入this\_addr，进行以下步骤，直到临时地址集为空：利用Ping判断该地址是否有效，若有效则将该地址存入有效地址集。Traceroute this\_addr，判断出this\_addr连接的路由器地址，然后利用启发式规则2猜测this\_addr所属的子网地址。  
**3.4 基于Ping和Traceroute的算法**　　算法的主要步骤是首先随机选取域内形式为\*.1的地址并将它们存入临时地址集，然后依次从临时地址集中取出地址存入this\_addr，进行以下步骤，直到临时地址集为空：利用Ping判断该地址是否有效，若有效则将该地址存入有效地址集并且根据启发式规则3将更多的地址加入临时地址集。Traceroute this\_addr，判断出this\_addr连接的路由器地址，然后利用启发式规则2猜测this\_addr所属的子网地址。  
**3.5 基于SNMP和ARP的算法**　　算法的主要步骤是将探测源模拟成一个网管站与SNMP Agent通信，先取得探测源的默认网关，存入待探测队列ToDoList。依次取出ToDoList中的IP，获得该IP的MIB库中ipRouteTable数据，当ipRouteType=direct时，利用ipRouteDest可以获得当前路由器连接的子网；当ipRouteType=indirect时，可以利用ipRouteNextHop获得路由器-路由器的连接，并将ipRouteDest存入ToDoList。获得ifToMediaNetAddress中的IP，判断其属于哪个子网。当ToDoList所有IP均被处理后，生成网络拓扑结构图。本算法与基于SNMP和Ping的算法的区别在于利用ARP表获得子网中的活动IP，因此提高了算法的速度，减轻了负载，但是却可能漏掉一些活动IP。  
**3.6 基于OSPF和Ping的算法**　　算法的主要步骤是先利用OSPF或RIP协议生成的路由信息获得路由设备以及子网间的连接关系，然后利用Ping获得子网中的活动主机的信息。  
**3.7 基于BGP和Traceroute的算法**　　算法的主要步骤是先利用BGP路由信息区分出Internet上的各个域，利用Ping获得每个域中的活动主机，Traceroute这些活动主机，利用路径信息结合BGP路由信息生成Internet主干网的拓扑信息。本算法主要用于发现Internet主干网的拓扑信息。  
　　表2对以上算法的优缺点进行了总结。

**4 网络拓扑发现算法的评价方法**　　(1)速度。可用算法执行所花费的时间来衡量。算法执行的时间分为两部分：采集信息生成拓扑结构的时间；将生成的表示拓扑关系的数据结构以图形化的形式显示出来的时间。  
　　(2)负载。因为一个算法中对网络造成的负载可能由多个部分引起，如在基于SNMP的算法中，给网络引入的负载包括获得拓扑信息的SNMP数据包和为判断一个地址是否有效所引入的ICMP报文(利用Ping引入的)。考虑到大部分拓扑发现算法中都会用到基于ICMP的工具(Ping、Traceroute)，并且由ICMP所引入的负载远远大于其他因素引入的负载，所以可以用一个算法用到的所有ICMP报文在网络中所经历的总跳数(hop)代表该算法对网络造成的负载[2]。按照这个定义，对于Ping，假设对每个节点Ping 2次，则要判断一个H跳处的设备是否有效所需要引入的负载是4H；对于Traceroute，假设对每个路由器发送两个探测包，这样对于一个在h(1≤h≤H)跳处的设备，所引入的负载是4h，则要Traceroute一个H跳处的设备所引入的负载是1到H的算术级数的4倍。  
　　  
　　(3)完整性。可用算法发现的网络设备数量占实际网络中设备数量的百分比表示。  
　　(4)准确性。可用算法面对多个可选的拓扑结构的可能性来表示。  
　　本文从协议、规则、算法及评价标准几个方面对网络拓扑发现技术进行了介绍，提出了量化评价网络拓扑发现算法的方法。为初步接触网络拓扑发现的人员提供了全面、详细的参考，同时也为产生新的网络拓扑发现算法提供基础研究。