ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.20, No.8, August 2009, pp.2280−2288
doi: 10.3724/SP.J.1001.2009.03398

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

别名解析中的别名过滤技术*

赵洪华+, 白华利, 陈 鸣, 魏镇韩

(解放军理工大学 指挥自动化学院,江苏 南京 210007)

Alias Filtering Technique in Alias Resolution

ZHAO Hong-Hua+, BAI Hua-Li, CHEN Ming, WEI Zhen-Han

(Institute of Command Automation, PLA University of Science & Technology, Nanjing 210007, China)

+ Corresponding author: E-mail: zhhahuatian@163.com

Zhao HH, Bai HL, Chen M, Wei ZH. Alias filtering technique in alias resolution. *Journal of Software*, 2009, 20(8):2280−2288. http://www.jos.org.cn/1000-9825/3398.htm

Abstract: To improve the efficiency of alias resolution in large scale network, a concept of alias filtering is proposed based on IP level topology measured by traceroute. The characteristics of alias relationship are explored theoretically, and three attributes are proposed to deal with traceroute data based on the characteristics. Then, an alias filtering algorithm called AF and a verification of alias filtering algorithm called VAR are put forward. Finally, both the algorithms are verified via the traceroute data from Internet covering China, Japan, and Korea, which are measured by Skitter of CAIDA (Cooperative Association for Internet Data Analysis). The results prove that the concept of alias filtering is very important and the algorithms proposed in this paper are valid and efficient.
Key words: alias resolution; network topology; IP address filter; algorithm

摘 要: 为了提高大规模网络中别名解析的效率,在用 traceroute 测量得到的 IP 级网络拓扑的基础上,提出别名过滤的概念.首先从理论上研究别名关系具有的性质,由此提出处理 traceroute 数据的 3 个属性;然后提出并设计了别名过滤算法 AF(alias filtering)和别名验证算法 VAR(validation of alias relationship).最后,利用 CAIDA(Cooperative Association for Internet Data Analysis)的 Skitter 项目得到的中国、日本、韩国这 3 个国家因特网的 traceroute 数据集对上述算法进行了验证分析.结果表明,别名过滤的概念非常重要并且文中提出的算法效率比较高.
关键词: 别名解析;网络拓扑;IP 地址过滤;算法
中图法分类号: TP393 **文献标识码**: A

随着因特网规模的飞速发展,网络结构也发生了巨大的变化,因特网已经成为人们日常生活不可缺少的一部分.为了更好地使用、管理和控制网络,必须要对网络的拓扑结构有充分的了解.

由于网络的异构性,要获得完整的网络拓扑结构比较困难.目前对于网络拓扑发现技术的研究比较多,其中以基于 traceroute 方式的研究最多.然而,基于 traceroute 方式只能发现 IP 级的网络拓扑,也称为逻辑拓扑,即从一台路由器入口的 IP 地址到下一台路由器入口的 IP 地址的连续序列,这种拓扑并未反映出路由器之间的真实连接情况.

* Received 2007-08-30; Revised 2008-02-02; Accepted 2008-06-11

1996年,Jack Rickard首先提出了利用traceroute来反映网络拓扑结构的思想.此后,对IP级网络拓扑的研究越来越多,也日趋成熟.1999年和2000年,Burch等人提出了基于单点测量网络拓扑的方法[1,2].他们选择了较多的目标节点.Govindan在此基础上又在Mercator[3]项目中采用了启发式算法来选择目标节点,并在测量中利用了源路由的方式.Skitter[4]是CAIDA(Cooperative Association for Internet Data Analysis)的基于traceroute的因特网拓扑发现项目,该项目从1998年开始研究,采用分布式测量的方式,在全球共设置了近20个测量源,测量目标节点近100万个.Rocketfuel[5]是由Neil Spring等人开发的另一个基于traceroute的、针对ISP(Internet service provider)的网络拓扑发现工具.该工具使用了全球480多个traceroute服务器.

与IP级网络拓扑相对的是路由器级网络拓扑,它反映了路由器之间的连接关系.这种拓扑更为网络管理、使用及维护人员所熟悉.在IP级网络拓扑的基础上,通过找出属于同一台路由器的IP地址集合,就能将IP级网络拓扑聚合成路由器级网络拓扑.所谓别名解析(alias resolution)则是这样一种找出属于同一台路由器的IP地址集合的技术.

当前,别名解析机制大致分为两类,即基于测量的方式和基于分析的方式.其中,基于测量的方式可分为以下两种方法:

(1) 基于测量数据分组的返回IP地址.在使用UDP(user datagram protocol)分组Ping一台路由器的IP地址时,返回的ICMP(Internet control message protocol)报文的源地址为路由器中距离测量源节点最近的端口IP地址,根据该返回地址则可判断两个IP地址是否属于同一路由器.Mercator[3]中的别名解析工具和iffinder[6]均采用了这种测量方法.

(2) 基于测量数据分组的返回报文ID号.由于路由器中有一个计数器来标识路由器处理过的报文,连续向路由器不同端口的IP地址发送Ping报文,然后观察返回报文的ID号是否相近,如果相近则它们属于同一台路由器.别名解析工具Ally[5,7]采用了这种测量方法.

上述两种方法只能用于验证两个IP地址是否属于同一台路由器,而且要求设备厂商没有对路由器作特殊限制性设计.

别名解析中基于分析的方式又可分为以下3种方法:

(1) 基于DNS(domain name system)信息分析IP地址.为了便于管理和维护,因特网服务提供商(ISP)对网络设备命名时通常采用DNS的规范,如同一台路由器的IP地址具有相同的域名,因此可以根据这种规范来分析某些IP地址是否属于同一台路由器.别名解析工具undns[5,8]采用这种原理进行判别.欧美地区对路由器的DNS命名非常规范,而亚太地区特别是中国和韩国对网络设备的DNS命名有所欠缺[5],使得该方法有时无法有效工作.

(2) 基于图论的分析.从图的角度,具有相同后继路由的两个IP地址属于同一台路由器[9].该方法在判定时往往不能依据实际网络情况的差异,例如路由器之间存在交换机或其他物理设备时,具有相同后继的IP地址属于同一台路由器的概率较小,所以基于图论的方式误判的几率较大.

(3) 基于对称路由.根据测量源IP和测量目标IP之间的两个方向的traceroute路径信息来确定同一台路由器的不同IP[10]地址.在大规模网络中,获得对称路由需要较多测量点的合作,因此这种方法在大规模网络中难以实施.

由于别名解析技术意在判断两个或多个IP地址是否属于同一台路由器的IP地址集合,因此,任意两个IP地址似乎都有可能需要进行别名解析.设IP地址的数量为N,如果要对任意两个IP地址解析,别名解析的次数为$N\times(N-1)/2$,为$O(N^2)$级.况且,每进行一次别名解析,为了获得准确结果都需要测量数次,而实际网络中需要判断的IP地址数量达到几万的场合是非常普遍的.盲目的别名解析所引发的测量流量将严重侵扰网络的正常运行,而所需要的时间也是不能容忍的.因此,别名解析技术还有一个重要问题需要解决,即在进行别名解析前,首先应当确定哪些IP地址需要解析,哪些IP地址无须解析,我们将其称为别名过滤技术(alias filtering technique).

本文从提高别名解析整体工作效率的角度出发,提出别名过滤的概念.第1节从理论上分析在来自traceroute测量的数据中需要进行别名解析的IP地址之间的关系,分析别名关系的性质.第2节根据相关理论设

计别名过滤算法 AF(alias filtering),在此基础上设计别名验证算法 VAR(validation of alias relationship).第 3 节用 CAIDA 测量的中国、日本、韩国这 3 个国家的 IP 级拓扑数据实际测实验证别名过滤算法 AF 和别名验证算法 VAR.第 4 节总结全文.

1 别名过滤的理论基础

1.1 别名的性质

定义 1(别名关系 *R*). 属于同一台路由器的两个 IP 地址具有别名关系,设一台路由器的所有 IP 地址集合为 A,$\forall ip_x\forall ip_y(ip_x,ip_y\in A\rightarrow ip_xRip_y)$.

性质 1. 别名关系 R 是一种等价关系.

证明:① R 满足自反性,即$\forall ip_x(ip_x\in V\rightarrow ip_xRip_x)$;

② R 满足对称性,即$\forall ip_x\forall ip_y(ip_x,ip_y\in V\wedge ip_xRip_y\rightarrow ip_yRip_x)$;

③ R 满足传递性,即$\forall ip_x\forall ip_y\forall ip_z(ip_x,ip_y,ip_z\in V\wedge ip_xRip_y\wedge ip_yRip_z\rightarrow ip_xRip_z)$. □

推论 1. 设 traceroute 测量的 IP 级拓扑数据集中的所有 IP 地址集合为 V,把每一个 IP 地址都作为集合 V 的子集,则所有 IP 地址组成 V 的最细划分.根据性质 1,别名关系是等价关系,则别名解析问题转换为划分问题.我们的工作即为找出集合 V 的最粗划分,每一个划分内部的 IP 地址都具有别名关系,即属于同一台路由器,通过对 IP 级拓扑数据根据别名关系作最粗划分即可发现路由器级拓扑结构.

定义 2(直接相连). 当两台路由器通过数据链路层链路相连时为直接相连,直接相连的路由器之间没有路由器、交换机或其他网络交换设备.

定义 3(traceroute 路径(简称路径)). traceroute 路径是指通过 traceroute 发现的从测量源 IP 地址到测量目的 IP 地址的 IP 地址序列.用 L_i 表示一条 traceroute 路径,$L_i=ip_{is}\rightarrow\ldots\rightarrow ip_{ix}\rightarrow\ldots\rightarrow ip_{id}$,用集合 V_i 表示路径 L_i 的节点集,$V_i=\{ip_{is},\ldots,ip_{ix},\ldots,ip_{id}\}$,用序偶$\langle ip_{ix},ip_{iy}\rangle$表示路径 L_i 中的一跳,其中称 ip_{iy} 为 ip_{ix} 的后继节点,用集合 E_i 表示路径 L_i 的链路(跳)集,$E_i=\{\langle ip_{is},ip_{i2}\rangle,\langle ip_{ix},ip_{iy}\rangle,\ldots\}$.

定义 4(发散路径). 发散路径是指具有相同测量源并且中间没有相交节点的路径.设测量源为 ip_s,路径 $L_i=ip_s\rightarrow\ldots\rightarrow ip_{ix}\rightarrow\ldots\rightarrow ip_{id}$,$L_j=ip_s\rightarrow\ldots\rightarrow ip_{jy}\rightarrow\ldots\rightarrow ip_{jd}$,其中,$ip_s$ 为测量源.L_i,L_j 是发散路径当且仅当

$$\forall ip_{ix}\forall ip_{jy}((ip_{ix}\in V_i\wedge ip_{jy}\in V_j\wedge ip_{ix}=ip_{jy})\rightarrow ip_{ix}=ip_{jy}=ip_s).$$

在研究 traceroute 路径中 IP 地址的别名关系时,我们假定下列 3 个条件:

条件 1(唯一连接性条件). 直接相连的两台路由器之间只有一条物理连接,设路由器 R_1 和路由器 R_2 相连,则存在 $ip_x\in R_1,ip_y\in R_2$,ip_x 和 ip_y 相连;如果$(\exists ip_w\in R_1)\wedge(\exists ip_z\in R_2)$,$ip_w$ 和 ip_z 相连,则 $ip_w=ip_x,ip_z=ip_y$.

条件 2(无环路条件). 在一条 traceroute 路径中,不存在循环路由.设路径为 $L=ip_1\rightarrow ip_2\rightarrow\ldots\rightarrow ip_x\rightarrow\ldots\rightarrow ip_y\rightarrow\ldots\rightarrow ip_d$,则$\forall ip_x\forall ip_y(ip_x,ip_y\in V\wedge ip_xRip_y\rightarrow ip_x=ip_y)$.

条件 3(单向唯一性条件). 在两台路由器之间的 traceroute 路径中,在一个方向上不存在两条或两条以上平行的 traceroute 路径.设路径经过路由器 R_1 到达路由器 R_2,路径 $L_i=ip_{i1}\rightarrow\ldots\rightarrow ip_{ix}\rightarrow\ldots\rightarrow ip_{iy}\rightarrow\ldots\rightarrow ip_{id}$,$L_j=ip_{j1}\rightarrow\ldots\rightarrow ip_{jx}\rightarrow\ldots\rightarrow ip_{jy}\rightarrow\ldots\rightarrow ip_{jd}$,$(ip_{ix},ip_{iy}\in R_1)\wedge(ip_{jx},ip_{jy}\in R_2)\rightarrow ip_{ix}Rip_{jx}\wedge ip_{iy}=ip_{jy}$.

当 traceroute 路径满足上述 3 个条件时,我们可以得到 3 个属性.这 3 个属性是我们的别名过滤技术的基础.

属性 1. 在一条 traceroute 路径中不存在别名关系.

证明:设路径为 $L=ip_1\rightarrow\ldots\rightarrow ip_x\rightarrow\ldots\rightarrow ip_y\rightarrow\ldots\rightarrow ip_d$,假设路径中存在两个 IP 地址具有别名关系,则有 $ip_xRip_y\wedge(ip_x)\neq(ip_y)$,这与条件 2 矛盾. □

属性 2. 发散路径之间不存在别名关系.

证明:设发散路径的起始点为 ip_s,路径集为 $L=\{L_1,\ldots,L_i,\ldots,L_j\}$,节点集为 $V=\{V_1\cup\ldots\cup V_i\cup\ldots\cup V_j\}$,边集 $E=\{E_1\cup\ldots\cup E_i\cup\ldots\cup E_j\}$.假设路径集中存在两个节点 ip_{ix} 和 ip_{jy} 分别为链路 L_i 和 L_j 的节点,$ip_{ix}Rip_{jy}\wedge ip_{ix}\neq ip_{jy}$,设 $ip_{ix}\in R_2\wedge ip_{jy}\in R_2$,令 ip_s 为路由器 R_1.根据条件 3,R_1 和 R_2 在方向 R_1 到 R_2 不存在两条或两条以上平行的路径,则 $ip_x=ip_y$,与假设矛盾. □

属性3. 具有相同后继 ip_c 的两个IP地址 ip_x 和 ip_y,如果 ip_x 和 ip_y 与后继 ip_c 直接相连则是别名关系,即 ip_xRip_y.

证明:令 ip_c 为路由器 R_c 的IP地址,即 R_c 的一个接口.ip_x 为路由器 R_x 的IP地址,ip_y 为路由器 R_y 的IP地址,R_x 和 R_c 之间有一条链路,R_y 与 R_c 之间有一条链路,而 R_c 只有1个接口与 R_x 和 R_y 连接,只能有1条连接,只有当 R_x 和 R_y 是一台路由器时才能满足条件,则 ip_xRip_y. □

上面3个属性在实际中比较直观,但当traceroute路径不能满足文中条件时,3个属性会出现异常情况,因此,在别名解析的过程中需要处理不能满足条件的traceroute路径.

1.2 traceroute路径中的别名情况

通过traceroute测量的是IP级网络拓扑.IP级网络拓扑中存在别名关系的情况归纳为3种.为明确起见,本文后面讨论别名关系 R 时,均指不同IP地址之间的别名关系.

(1) IP级拓扑中的traceroute路径存在对称路段的情况.设两条路径 L_i,L_j,路径 $L_i=ip_{is}\rightarrow\dots\rightarrow ip_{ix}\rightarrow ip_{iy}\rightarrow ip_{iz}\rightarrow ip_{iw}\rightarrow\dots\rightarrow ip_{id}$,路径 $L_j=ip_{js}\rightarrow\dots\rightarrow ip_{jx}\rightarrow ip_{jy}\rightarrow ip_{jz}\rightarrow ip_{jw}\rightarrow\dots\rightarrow ip_{jd}$,其中,$ip_{iy}Rip_{jz}$,$ip_{iz}Rip_{jy}$(如图1所示),定义这种情况为对称别名.判断两条路径是否有对称路段,可以根据两条路径中IP地址的网络标识,在网络中两个直接相连的路由器的端口IP地址通常具有相同的30比特网络标识[9].

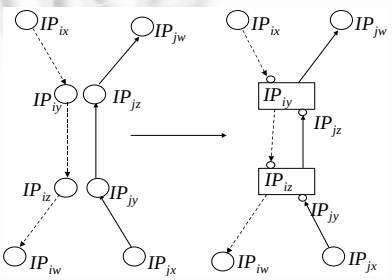

Fig.1 Symmetric alias relation in IP topology

图1 IP级拓扑中的对称别名关系

(2) IP级拓扑中的traceroute路径存在相同后继的情况.两条路径 $L_i,L_j,(\exists\langle ip_x,ip_y\rangle\in E_i)\wedge(\exists\langle ip_w,ip_z\rangle\in E_j)(ip_z=ip_y,ip_x\neq ip_w)\rightarrow ip_xRip_w$,定义这种情况为三角别名(如图2所示).

(3) IP级拓扑中的两条平行路径存在别名的情况.traceroute路径 $L_i=ip_{is}\rightarrow\dots\rightarrow ip_{ix}\rightarrow\dots\rightarrow ip_{id}$,$L_j=ip_{js}\rightarrow\dots\rightarrow ip_{jy}\rightarrow\dots\rightarrow ip_{jd}$,存在 $ip_{ix}\in E_i,ip_{jy}\in E_j,ip_{ix}Rip_{jy}$,定义这种情况为平行别名(如图3所示).

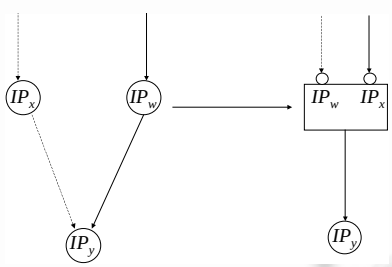

Fig.2 Triangle alias relation in IP topology

图2 IP级拓扑中的三角别名关系

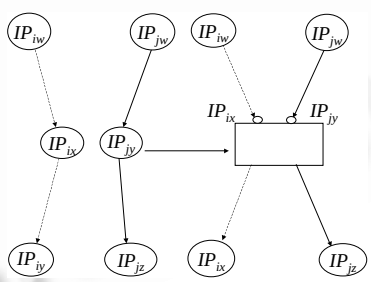

Fig.3 Parallel alias relation in IP topology

图3 IP级拓扑中的平行别名关系

2 别名过滤算法和别名验证算法

我们的别名解析工作主要是应用在规模较大的IP级拓扑数据上.当IP地址数量非常少时,可采用其他别名解析算法.别名解析工作主要分为两部分:① 对IP级拓扑中的IP地址进行别名关系过滤.在所有IP地址集合中过滤出可能具有别名关系的IP地址,排除掉不可能具有别名关系的IP地址,为此设计实现了别名过滤算法AF;② 对过滤后别名关系的验证.由于步骤①只找出了可能具有别名关系的IP地址集合,为了进一步确认过滤出的IP地址是否具有别名关系,提高别名解析的准确性,设计实现了别名关系验证算法VAR.

2.1 别名过滤算法AF

别名过滤分为两部分:第1部分根据三角别名和对称别名的性质过滤出所有可能具有三角别名和对称别名的IP地址集合;第2部分是过滤出所有可能具有平行别名关系的IP地址集合.

三角别名和对称别名都具有规律性,根据三角别名和对称别名的特征进行过滤.过滤的基本思想是:① 通

过扫描所有的 traceroute 路径,把所有具有相同后继的 IP 地址划分在一个集合内;② 通过扫描所有的 traceroute 路径,把后继具有相同 30 位网络地址[9]的 IP 地址划分在一个集合之内.在三角别名过滤和对称别名过滤过程中,根据别名关系的传递性把具有交集的 IP 地址集合合并为一个 IP 地址集合.

为了提高别名过滤的效率,每一个节点用一个六元组表示(*num*,*sour*,*local*,*next*,*des*,*rttl*).其中,*num* 表示路径号,*sour* 表示测量源,*local* 表示本跳 IP,*next* 表示下一跳 IP,*des* 表示目标节点,*rttl* 表示距离目标节点 *des* 的跳数.为了便于计算,IP 地址采用整数表示,并且在 IP 地址的查找和比较过程中采用了哈希技术.

三角别名和对称别名的过滤算法初始时把每一个 IP 地址作为一个集合,经过三角别名和对称别名的过滤找出所有可能具有三角别名关系或对称别名关系的 IP 地址进行集合的合并操作,算法的形式化描述见算法 1.

算法 1.

输入:指定地区 IP 级拓扑数据,拓扑数据中路径集为 L,节点集为 V.

输出:节点集 V 的最粗划分.

初始化:$V=\varnothing$

For (every $L_i \in L$) Do

Begin

While (L_i 没有分析完) do

Begin

从 L_i 中取出节点 v_x,根据 L_i 的信息生成六元组 s_x,生成集合 $X=\{v_x\}$;

把六元组 s_x 插入数据库和 hash 表;

If ((hash 表中存在六元组 s_y)∧($s_y.next=s_x.next$)∧({$s_y.$}∩{$s_x.pre$}=∅)∧({$s_y.$}∩{$s_x.next$}=∅)) then

Begin

s_y 的集合 $Y=Y\cup X$;

更新 V 中的集合 Y;

Continue;

End

else if (v_x 不是最后一跳∧hash 表中存在六元组 s_y,$s_y.next=s_{y+1}$ 与 $s_x.next=s_{x+1}$ 具有相同的 30 位网络号∧{$s_{y+1}.pre$}∩{$s_{x+1}.pre$}=∅) then

Begin

令 v_a 表示 s_x 的前一跳,设 v_a 的集合 A,v_y 的集合为 Y;

if $\exists v_t(v_t \in Y \wedge v_t R v_a)$ then

Begin

$V=(V-A-Y)$;

$Y=A\cup Y$;

把 Y 加入到集合 V;

令 v_b 表示 s_y 的前一跳,设 v_b 的集合 B;

$V=(V-B)$;

$B=B\cup X$;

把 B 加入到集合 V;

End;

End;

End

End;

上述算法找出了对称别名和三角别名的两种情况.AF 算法的第 2 步是发现 traceroute 路径中所有可能具有平行别名关系的 IP 地址集合,平行别名关系不同于三角别名和对称别名关系,平行别名关系没有固定的规律可循,因此我们采用排除方法.

平行别名关系的初始集合为经过三角别名和对称别名过滤后的所有 IP 地址集合,根据第 1.1 节的属性,排除所有不可能具有别名关系的 IP 地址,把可能具有别名关系的 IP 地址划分到一个集合.我们采用以下原则来过滤不可能具有别名关系的 IP 地址:

(1) 根据属性 1,一条 traceroute 路径中不存在别名关系,对于每个 IP 地址,排除同一路径中的其他 IP 地址.

(2) 根据属性 3,同一测量源的发散路径中的节点没有别名关系.设基于测量源 s 的路径集为 L_s,AF 算法的第 1 步所找到的具有别名关系节点的路径集为 L_1,则发散路径 $L=L_s-L_1$.

(3) 除了边界节点以外,不属于同一个 ISP 的节点没有别名关系,根据 ICANN(Internet Corporation for Assigned Names and Numbers)发布的 IP 信息,分配文件可以获得指定地区内不同 ISP 的 IP 地址段.

(4) 距离相同目标节点的 *rttl* 相差大于 3 的节点不存在别名关系.我们认为,如果到达同一目标节点的路径中的两个 IP 地址 ip_1,ip_2 属于一台路由器,则这两个 IP 地址与目标节点的 *rttl* 相差不大于 3.

不同地区的节点间不存在别名关系.通过从 ICANN 获得的 IP 地址信息和从 ISP 获得的 IP 地址信息,可以通过地理关系对 IP 作划分,划分成不同地区的 IP 地址集合,还可以采用经验原则推断节点所属的城市[10].

通过三角别名、对称别名和平行别名关系的过滤算法,把初始输入的 IP 地址集合作了划分,每个划分内部的 IP 地址都有可能具有别名关系.为了进一步确定划分内部的别名关系,我们通过别名验证算法来验证.

2.2 别名验证算法VAR

因为别名过滤是找出所有可能具有别名关系的 IP 地址集合,因此需要进一步确认.别名验证 VAR 算法在 AF 算法的基础上,对通过 AF 算法合并的 IP 地址集合采用基于返回 IP 报文的 ID 号的方式作别名验证,排除别名过滤中误判的情况.VAR 算法对每个划分内部的任意两个 IP 地址作别名关系验证,把经验证后具有别名关系的 IP 地址划分到一个集合中,把不响应的 IP 地址划分到一个集合.别名验证算法 VAR 的形式化描述见算法 2.

算法 2.

输入:AF 的结果集 $V=\{V_1,\ldots,V_i,\ldots,V_j\}$.

输出:确认后的别名关系 $V_R=\{V_{1R},\ldots,V_{iR},\ldots,V_{jR}\}$.

初始化:$V_R=\varnothing$,

```
For (every Vi∈V) Do
    Begin
        n=|Vi|;
        for (i=1;i≤n;i++)
        begin
          if Vi[i]标记为不响应  then continue;
          for (j=i;j≤n;j++)
          begin
            if Vi[j]标记为不响应, then continue;
            对 Vi[i],Vi[j]作测量;
            if Vi[i] not response then
            begin
              标记 Vi[i]为不响应,把 Vi[i]加入不响应的集合;
              Break;
            end;
            if Vi[j] not response then  标记 Vi[j]为不响应;
            if Vi[i]和 Vi[j]为别名关系(|IPID1−IPID2|≤100), then 加入到 Vi[i]的别名关系集合;
          end;
        end;
    End;
```

2.3 算法性能分析

AF 算法在过滤过程中需要分析所有的 IP 地址,把所有 IP 地址存储在内存中,因此所需的空间复杂度为 $O(N)$;AF 算法在过滤时每遇到一个新的 IP 地址,都会与已经存在的 IP 地址作别名过滤的查找和比较.我们采用 hash 函数作查找和比较运算,则每个 IP 地址查找和运算的时间为常数,设为 k,则 N 个 IP 地址所需的时间为 Nk,因此,AF 算法的时间复杂度和空间复杂度均为 $O(N)$.通过 AF 算法,过滤出所有可能具有别名关系的 IP 地址,把

可能具有别名关系的 IP 地址组成集合.

VAR 算法是在 AF 算法的基础上进行的,对每个集合中的任意两个 IP 地址作别名验证.由于 AF 算法已经过滤掉大量的 IP 地址,并对有可能存在别名关系的 IP 地址作了划分,因此别名验证时产生的附加流量要远远小于未作 AF 算法时产生的附加流量.设 AF 算法过滤出的 IP 地址数量为 m,当 m 个 IP 地址组成一个集合时,别名验证次数和产生的流量最大,流量 $flux=m\times(m-1)/2$,为 $O(m^2)$级;当 m 个 IP 地址组成 $m/2$ 个集合,每个集合只有 2 个 IP 地址时,别名验证次数和产生的流量最小,流量 $flux=m/2$,为 $O(m)$级.由于 m 远远小于 N,因此经过 AF 算法,别名验证次数和所产生的附加流量大大减少.

如果采用其他别名解析算法,在同样数量的 IP 地址的别名解析过程中,iffinder 和 Ally 算法的复杂度都为 $O(N^2)$,并且发送的附加流量和使用的时间均为 $O(N^2)$级,而采用 undns 和对称别名解析方法则无法获得大量的别名解析结果.

对比其他别名解析的过程,我们对别名的处理结合上述两个算法,具有如下优点:

(1) 通过别名关系的过滤算法,提前选择出可能具有别名关系的 IP 地址和过滤掉不可能具有别名关系的 IP 地址,大幅度压缩了需要进行别名解析的节点数量,提高了别名解析的效率.

(2) 综合使用了别名解析中基于测量的方式和基于分析的方式,结合了两者的优点.

(3) 尽最大努力找出了所有别名关系的情况,比现有的其他方法发现的别名关系更全面.

(4) 在三角名别名关系上应用了验证算法,避免了三角别名关系中误判现象的产生.

(5) 在对称别名关系上应用了验证算法,避免了对称别名关系中误判情况的发生.

与其他别名解析方法类似,也会存在别名关系漏判和误判的情况:

(1) 漏判的出现有 3 种情况:① 对称别名的漏判.当具有对称别名关系的两个 IP 地址的后继 IP 没有相同的 30 为网络号[9]时会出现漏判的情况.② 平行别名的漏判.当 traceroute 路径不符合条件 3 时会存在漏判的情况.这两种情况在其他别名解析算法中都不能排除,要排除这两种情况只有对所有的 IP 地址作两两验证,当 IP 地址数量较多时,这是不现实的.③ 在别名验证时,如果路由器不响应,则会出现漏判的情况,这是基于测量方式的别名解析都会遇到的问题.

(2) 由于别名验证是采用基于返回 IP 报文的 ID 号的方式,当具有别名关系的 IP 地址返回报文的 ID 号不连续时会出现误判的情况,一般连续的范围为 100,路由器的计数器为 32 位.两台路由器的 ID 号如果连续则会出现误判,误判的概率为 $100/2^{32}=2.3\times10^{-8}$,此值几乎为 0.所以,我们的别名解析过程误判的概率很小,与算法 Ally 误判的概率相同.

经过上面的对比分析可知,我们的别名解析过程与现有别名解析算法相比极大地提高了别名解析效率(在大规模 IP 级拓扑数据上,其他方法无法正常工作),在别名解析过程中尽最大努力发现了所有的别名关系,在别名解析结果中出现误判的概率很小,可以忽略.

3 算法的验证

3.1 获取IP级网络拓扑数据

为了对上述算法进行验证,我们从 CAIDA 的 skitter 项目获得 IP 级网络拓扑数据.该数据是 CAIDA 对全球因特网做 traceroute 测量所获得的真实数据.由于每天得到的最新测量数据都有几百 MB 之多,我们必须首先选取我们感兴趣的 IP 级网络拓扑数据.其基本步骤如下:

(1) 从 skitter 的测量服务器下载最新的数据文件.

(2) 从 ICANN 下属 5 个区域性 IP 地址分配机构 ARIN(American Registry for Internet Numbers), RIPENCC (Réseaux IP Européens Network Coordination Center),APNIC(Asia-Pacific Network Information Center), LACNIC(Latin American and Caribbean Internet Addresses Registry)和 AFRINIC(Africa Internet Addresses Registry)获取有关国家或地区的最新地址分配文件.

(3) 对每条 skitter 数据记录根据国家或区域信息提取相关的 traceroute 路径.一条 skitter 数据记录包含 14

个字段,其中 3 个字段含有地址信息,分别是源 IP、目的 IP 和中间各跳的 IP 地址序列,这 3 个字段表示了一条完整的 traceroute 级路径.

(4) 分解每条区域 traceroute 路径形成节点和链路信息,把区域内的路径信息分解为节点和链路信息.

由于这样得到的 traceroute 路径仍然不能完全满足第 1 节的 3 个条件,因此我们采用下面的原则来处理特殊情况的 traceroute 路径.

(1) 对于目标地址不可达的 traceroute 路径,只保留所发现的 traceroute 路径部分.设测量源为 ip_s,测量目标地址为 ip_d,traceroute 发现的最后一跳 ip_x,如果 $ip_x \neq ip_d$,则只保留路径 $ip_s \rightarrow \ldots \rightarrow ip_x$.

(2) 对于 traceroute 路径中存在匿名路由器的情况,删除匿名路由器的部分.设路径 $L=ip_s \rightarrow \ldots \rightarrow ip_x \rightarrow *** \rightarrow ip_y \rightarrow \ldots \rightarrow ip_d$,其中 ip_x 和 ip_y 之间为匿名路由器,则只保留路径 $ip_s \rightarrow \ldots \rightarrow ip_x$ 和 $ip_y \rightarrow \ldots \rightarrow ip_d$.

(3) 对于存在循环路由的 traceroute 路径,删除循环部分的链路.① 对于一条路径中存在循环路由的情况,设 $L_i=ip_{is} \rightarrow \ldots \rightarrow ip_{ix} \rightarrow \ldots \rightarrow ip_{ix} \rightarrow ip_{iy} \rightarrow \ldots \rightarrow ip_{id}$,则只保留路径 $L_i=ip_{is} \rightarrow \ldots \rightarrow ip_{ix} \rightarrow ip_{iy} \rightarrow \ldots \rightarrow ip_{.id}$;② 对于两条路径中存在循环路由的情况,设 $L_i=ip_{is} \rightarrow \ldots \rightarrow ip_x \rightarrow ip_y \rightarrow \ldots \rightarrow ip_{id}$,$L_j=ip_{js} \rightarrow \ldots \rightarrow ip_y \rightarrow ip_x \rightarrow \ldots \rightarrow ip_{jd}$,存在路径 $ip_x \rightarrow ip_y$ 和路径 $ip_y \rightarrow ip_x$,只保留 $ip_x \rightarrow ip_y$ 或 $ip_y \rightarrow ip_x$.

(4) 对于具有欧几里德性质的路径,只保留最长的一部分.设路径中存在 $ip_x \rightarrow ip_y \rightarrow ip_z$ 和 $ip_x \rightarrow ip_z$,则只保留 $ip_x \rightarrow ip_y \rightarrow ip_z$ 部分.

3.2 别名解析结果分析

首先对 Skitter 项目在 2007 年 7 月 16 日发布的 traceroute 数据集作了处理,获取中国大陆、韩国和日本的 IP 级拓扑数据.其中,中国大陆、韩国和日本的 IP 级拓扑数据集中包含的 IP 地址数量分别为 29 131,46 936 和 47 516.我们首先应用 AF 算法过滤出每个国家的数据集中可能具有别名关系的 IP 地址集合,对经过 AF 过滤后每个地区的 IP 地址集再用 VAR 算法作了 3 次别名关系的验证.由于不同时间网络状况的不同,我们取其中响应节点数量最多的一次作为结果.我们的分析结果包括从 Skitter 获取的 3 个国家 IP 级拓扑的 IP 地址数量、经确认的具有别名关系的 IP 地址数量及所占的比例和在确认别名关系时不响应的 IP 地址数量、别名解析所需要的时间等(见表 1).在表 1 中,不响应的 IP 地址在别名关系过滤时根据过滤的原则可能具有别名关系,但在进行验证时并不响应测量报文.路由器不响应别名解析的原因多种多样,对安全方面的考虑可能是最主要的原因,也有可能是路由发生而变化而原有的 IP 地址不再使用.

Table 1 Alias resolution results of China, Japan and Korea

表 1 中国、日本、韩国的别名解析处理结果

Country	Deal time	IP count	IP count of alias relation/ratio	IP count of unresponsed/ratio
China	7h04m	29 131	2 412/(8.28%)	3 360/(11.53%)
Korea	4h41m	46 936	1 666/(3.55%)	4 655/(9.92%)
Japan	5h12m	47 516	1 665/(3.50%)	3 818/(8.04%)

由表 1 可知,中国、日本、韩国这 3 个国家的 IP 地址数量都有几万个,由于在 3 个国家中路由器的 DNS 命名很少[5],所以基于 DNS 解析的方式不适用.

在别名解析过程中,基于图论的方式只能发现具有三角别名关系的 IP 地址,不能找出具有对称别名关系和平行别名关系的 IP 地址,并且没有验证.我们的别名解析与基于图论的方式相比:首先找出可能具有三角别名关系的所有 IP 地址并进行了验证,所以误判率要远远小于基于图论的方式;其次,我们的别名解析还找出了具有对称别名关系和平行别名关系的集合,比基于图论的方式发现的别名关系要多.

我们的算法与基于返回 IP 地址的 iffinder 和基于返回报文 ID 号的 Ally 相比具有较高的效率,iffinder 和 Ally 都是对两个 IP 地址进行别名关系判定,当 IP 地址数量较多时,工作量非常大.以 IP 地址数量最少、我们的算法所用时间最长的中国 IP 级拓扑为例,如果 iffinder 和 Ally 要达到同样的效果,需要对任意两个 IP 地址作别名解析,假设 iffinder 或 Ally 每次对一个 IP 地址作别名解析探测所用的时间为 10ms(实际验证时间通常要大于此时间),设所需的总时间为 T,

$$T=\frac{29131\times 29130\times 10\times 2}{2\times 1000\times 60\times 60}=2357(\mathrm{h})\approx 98(\mathrm{D}).$$

如果以实际测量过程中网络节点不响应的时间和实际测量时间计算,则所用时间还会大大增加,同时由于测量的次数为 $O(N^2)$级,所以测量产生的附加流量更是不可估量的.

由表 1 可知,中国的 IP 级网络拓扑中具有别名关系的 IP 地址最多,确认的和非确认的具有别名关系的 IP 地址数量都大于日本和韩国.例如,已确认的具有别名关系的IP地址所占比例,中国为日本和韩国的2倍多,而不响应的 IP 地址所占比例也大于日本和韩国.在已确认的别名关系中,别名关系集合所包含的 IP 地址数量不同.例如,在韩国的IP地址中,最大的一个别名关系集合含有24个IP地址,而中国和日本分别为6个和8个.若所有的 IP 地址都响应,则结果还会更好.当然,上述分析结果根据测量方法和被测目标对象的不同也会有所不同.

4 结束语

为了提高别名解析过程的效率,本文提出了别名过滤的思想.在严格的理论分析基础上,提出了一种别名过滤算法 AF 和别名验证算法 VAR.通过来自 CAIDA 的 Skitter 项目测量的中国、日本和韩国 3 个国家的 IP 级拓扑数据集进行的验证分析,表明了别名过滤技术的有效性,也表明了利用本文算法可以高效地从 IP 级网络拓扑得到路由器级的网络拓扑.

References:

[1] Danesh A, Trajkovic L. Mapping the Internet. IEEE Computer, 1999,32(4):97−98, 102.

[2] Cheswick B, Burch H, Branigan S. Mapping and visualizing the Internet. In: Proc. of the USENIX Annual Technical Conf. San Diego, 2000. 1−12. http://citeseer.ist.psu.edu/451798.html

[3] Govindan R, Tangmunarunkit H. Heuristics for Internet map discovery. In: Sidi M, Sengupta B, eds. Proc. of the IEEE INFOCOM 2000. Tel Aviv: IEEE Press, 2000. 1371−1380.

[4] Claffy K, Monk T, McRobb D. Skitter. 1999. http://www.caida.org/tools/measurement/skitter/

[5] Spring N, Mahajan R, Wetherall D, Anderson T. Measuring ISP topologies with Rocketfuel. IEEE/ACM Trans. on Networking, 2004,12(1):2−16.

[6] Daniel M, Claffy K. Iffinder. 2000. http://www.caida.org/tools/measurement/iffinder/

[7] Neil S. Ally. 2005. http://www.cs.washington.edu/research/networking/rocketfuel/

[8] Neil S. Undns. 2005. http://www.cs.washington.edu/research/networking/rocketfuel/

[9] Spring N, Dontcheva M, Rodrig M, Wetherall D. How to resolve IP aliases. 2004. http://www.cs.washington.edu/homes/rodrig/pubs/alias_res.pdf

[10] Gunes M, Sarac K. Analytical IP alias resolution. In: Proc. of the IEEE Int'l Conf. on Communications (ICC). 2006. http://www.utdallas.edu/~ksarac/research/publications/ICC06-G.pdf

赵洪华(1979－),男,河北吴桥人,博士,讲师,主要研究领域为网络测量,网络管理,分布式计算.

陈鸣(1956－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为计算机网络,网络测量,分布式计算.

白华利(1981－),男,博士生,助理工程师,主要研究领域为网络拓扑发现.

魏镇韩(1976－),男,博士生,工程师,主要研究领域为网络测量,分布式计算.