第35卷 第20期 计 算 机 工 程 2009年10月
Vol.35 No.20 Computer Engineering October 2009
·网络与通信· 文章编号：1000—3428(2009)20—0117—04 文献标识码：A 中图分类号：TP311.52

基于IP地址指派惯例的路由器别名识别方法

胡 博，陆余良，杨国正，夏 阳
(解放军电子工程学院网络工程系，合肥 230037)

摘 要：路由器别名识别是构建路由器级网络拓扑的重要步骤之一。现有的别名识别工具大多采用主动探测方法，给网络带来额外负荷并依赖路由器的响应。针对该问题，提出一种新的路由器别名识别方法ABR，该方法利用通用IP地址分配方案对traceroute工具探测得到的路径进行别名识别。实验结果表明，与传统方法相比，该方法的识别正确率和效率均有显著提高。

关键词：拓扑发现；路由器别名；IP地址指派

Method of Router's Alias Recognition Based on IP Address Assignment Practice

HU Bo, LU Yu-liang, YANG Guo-zheng, XIA Yang
(Network Engineering Department, Electronic Engineering Institute, Hefei 230037)

【Abstract】 Router's alias recognition is an important step in generating router level topology. Existing tools for alias recognition use an active probing approach, they induce traffic overhead into the network and depend on the response of the routers. Aiming at this problem, this paper proposes a novel approach called ABR. The method utilizes the common IP address assignment scheme to infer router's alias from the path traces collected by traceroute tool. Experimental results show that the new method significantly improves the precision and efficiency compared to existing approaches.

【Key words】 topology discovery; router's alias; IP address assignment

1 概述

获取网络的拓扑结构是Internet网络测量的研究基础，现有路由器级网络拓扑发现技术主要采用基于traceroute工具的探测结果构建网络拓扑图。构建过程中的一个重要步骤就是需要确定在traceroute返回的路由器接口IP地址中，哪些接口属于同一个路由器，即路由器别名识别问题[1]，路由器别名识别的目标就是合并属于同一路由器的多个接口IP地址。

路由器别名问题对所构建的路由器级网络拓扑的正确性影响较大，它使所构建的网络拓扑图包含虚假节点和链路。现有的别名识别方法大都是建立在路由器对探测包产生响应基础上的主动探测算法，这类算法受到路由器配置的制约。针对现有方法的不足，本文提出一种新的基于IP地址分配惯例的别名识别方法ABR。

2 相关研究

目前，对路由器别名识别主要采取主动探测的方式。文献[2]提出通过比较路由器操作系统返回包中的源地址字段来识别别名的算法，通过比较ICMP端口不可达报文的源地址字段和原始探测目的地址字段来识别路由器别名。Mercator是基于该算法开发的别名识别工具，它的不足是当2个IP地址位于路由器同一端时，无法正确判断别名情况。

文献[3]提出一种别名识别的新工具(Ally)，它基于源地址识别，并增加了对ICMP响应报文中IP identifier字段的检查。IP identifier字段原本用于数据包的分段重组。该字段的计数随着每一个数据包的发送而线性递增。该特性使从一个路由器连续发出的数据包拥有连续的IP identifier计数，Ally通过观察这一特性来判断路由器别名。它的主要问题是对节点数为 n 的网络进行别名识别，须对 $O(n^2)$数量级的IP地址对进行测试。

3 基于IP地址指派惯例的别名识别方法

上述主动探测算法均利用ICMP差错报文来推断路由器的别名，算法假设路由器都会产生ICMP响应报文，这类方法简单实用。但不同路由器厂商对ICMP响应机制的不同设置，以及网络管理员对路由器的不同配置都会对这类算法的效果产生制约[4]。本文提出的基于IP地址指派惯例的别名识别方法是一种被动分析算法，它通过分析各接口IP地址，依据IP地址指派惯例识别路由器之间的点对点连接，合并路由器别名。

3.1 IP地址指派惯例

IP地址空间的分配遵循IP地址空间注册向导[5]。路由器点对点链路上的IP地址处于同一子网内，由于点对点链路只有2个网络接口，因此子网掩码为/30最适合点对点链路上的子网划分，最后2 bit刚好可以标识2个IP地址，这样的指派使该子网的IP地址利用率达到最高，以尽可能节约IP地址资源。根据文献[6]，子网掩码为/31的网络地址也开始用

作者简介：胡 博(1983－)，男，硕士研究生，主研方向：计算机网络安全；陆余良，教授、博士生导师；杨国正，博士研究生；夏 阳，博士

收稿日期：2009-02-01 **E-mail：**jsnjhb@126.com

于路由器之间的点对点连接。这种分配上的特殊性可以用来推断 2 个端系统之间 traceoute 路径上的点对点链路，也是本文所提出路由器别名识别算法的依据。

3.2 点对点链路上的别名识别

当给出已得到的多条 traceroute 路径时，须在这些路径中寻找这种掩码分配特殊性，以此推断路由器别名。IP_A 是其中一条路径上的 IP 地址，IP_B 是另一条路径上的 IP 地址，当 IP_A 和 IP_B 满足/30 或/31 的连续性时，就找到了一个匹配，记为 $IP_A \xleftrightarrow{30} IP_B$ ($IP_A \xleftrightarrow{31} IP_B$)，说明这 2 条路径包含一段点对点链路，是 2 条路径的公共部分。综合这些匹配，即可将这 2 条路径上所有满足这一匹配的部分合并为公共的对称路径。在这条对称路径上，原本分散在 2 条路径上的路由器接口 IP 将作为同一路由器出现。

下文通过一个例子来说明如何通过路由器点对点链路上 IP 地址分配特殊性识别路由器别名，须注意终端主机与第一个相连的路由器也有可能是点对点链接的情况，应排除所有终端主机。

如图 1 所示，h_1,h_2,h_3,h_4 是终端主机，r_1,r_2 是它们之间的路由器，r_1 与 r_2 之间通过点对点链路连接。小写字母 $a,b,\cdots,j$ 表示接口 IP 地址。假设已经得到由 traceroute 获得的 2 个接口 IP 列表 *trace*(*h*1, *h*4)=(*a, b, j, h*)和 *trace*(*h*2, *h*3)=(*c, i, e, f*)，通过搜索这 2 个路径，寻找满足/30 或/31 子网掩码类型的匹配，若找到就说明这 2 条路径有公共的点对点链路，可以进行点对点链路上别名识别。

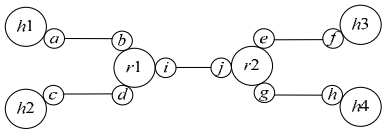

图 1 路由器与接口关系

接口匹配的 1 种情况如图 2 所示，从中可以得到一个匹配 $i \xleftrightarrow{30} j$，接口 *i* 和 *j* 处于同一子网并且该子网只有 2 个可分配的 IP 地址，不存在第 3 个接口，因此，*i* 和 *j* 各自所处的路由器之间的子网只构成唯一的一条点对点链路，在这个链路 2 端只有 2 个直连的路由器，它们之间不存在第 3 个路由器。

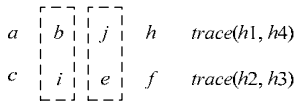

图 2 接口匹配的 1 种情况

该网络的不同路径中凡是出现 *i* 或 *j* 的地方(由 traceroute 的原理可知 *i, j* 不会同时出现在一条路径中)，*j* 的上一跳与 *i* 为同一个路由器上的接口，*i* 的下一跳与 *j* 为同一个路由器上的接口。因此这里识别出路径 *trace*(*h*1, *h*4) 中 *j* 的上一跳 *b* 和接口 *i* 为一对路由器别名 (*b, i*)，同理，路径 *trace*(*h*2, *h*3) 中 *i* 的下一跳 *d* 和接口 *j* 为一对路由器别名 (*j, e*)。

接口匹配的另 1 种情况如图 3 所示，推断出的结果为别名对 (*j, c*) 和 (*h, i*)。

由文献[7]可知，在默认情况下，路由器发送 ICMP 差错报文根据最短路径原则选路。接口 *j* 由 h_1 探测得到，所以与接口 *j* 同处一个子网的接口 *i* 也处于靠近 h_1 的一端，因此，图 3 所示的情况可以被排除。

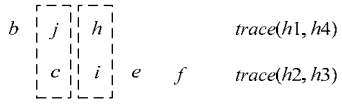

图 3 接口匹配的另 1 种情况

一个实际的情况，选定 2 个主机 *h*1, *h*2，IP 地址分别为 218.185.222.156 和 61.241.130.154，分别从一台主机向另一台主机进行 traceroute 探测，从 *h*1 到 *h*2 和从 *h*2 到 *h*1 的探测路径如图 4、图 5 所示。

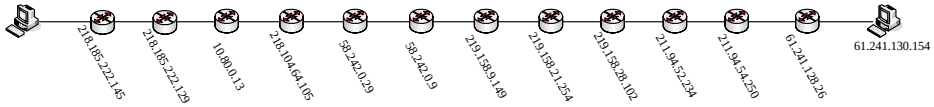

图 4 *h*1->*h*2 的探测路径

图 5 *h*2->*h*1 的探测路径

由于从 *h*1->*h*2 的探测路径和 *h*2->*h*1 的探测路径并不是同一条路径，因此不能简单地将 2 个方向的路径进行合并。

图 4 和图 5 中出现了多对相邻 IP 地址的情况，根据 IP 地址指派惯例中对路由器点对点连接时采用/30 或/31 掩码的原则，可以判定，2 条路径上出现连续性 IP 地址的路由器之间是点对点连接。表 1 给出了路径的每一跳信息，其中 *h*2->*h*1 的路径是反向路径。

表 1 *h*1 和 *h*2 之间 traceroute 结果

Hop	*h*1->*h*2(正向路径)	*h*2->*h*1(反向路径)
1	218.185.222.145	Request timed out
2	218.185.222.129	218.185.222.130
3	10.80.0.13	Request timed out
4	218.104.64.105	218.104.64.106
5	58.242.0.29	58.242.0.30
6	58.242.0.9	58.242.0.10
7	219.158.9.149	219.158.9.2
8	219.158.21.254	219.158.21.253
9	219.158.28.102	219.158.28.101
10	211.94.52.234	211.94.52.233
11	211.94.54.250	211.94.54.249
12	61.241.128.26	61.241.128.25
13	61.241.130.154	61.241.130.190

路径中的第 2 跳、第 4 跳~第 6 跳、第 8 跳~第 12 跳都出现了连续的 IP 地址。这些 IP 地址对符合路由器连接中点对点连接掩码为/30 或/31 的条件，可认为地址 218.185.222. 129 和 218.185.222.130 是点对点连接，218.104.64.105 和 218.104.64.106 是点对点连接。通过反复进行点对点链路上的别名识别，合并同一路由器的不同接口，可以构造如图 6 所示的对称路径。

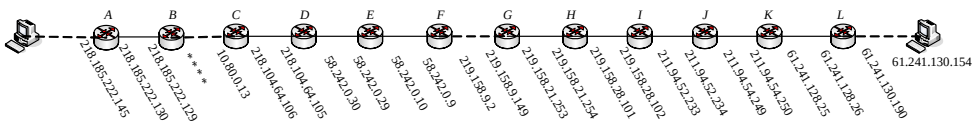

图 6 合并后的对称路径

其中，路由器之间的实线连接表示点对点连接，虚线表示连接情况未知。观察图 6 的路径可以看出整个链路的路由器别名情况：IP 地址 218.185.222.145 和 218.185.222.130 是 *A* 的 2 个接口，10.80.0.13 和 218.104.64.105 是路由器 *C* 的 2 个接口等。

3.3 算法描述

基于 IP 地址指派惯例的路由器别名识别算法使用上述利用连续性 IP 地址推断点对点连接的思想，分析给定源点与目的节点之间的路径对。

定义 1 当 IP_A 和 IP_B 属于同一个/30 或/31 掩码的子网时，逻辑式 $IP_A \leftrightarrow IP_B$ 为真。其中，$\leftrightarrow$ 为一逻辑运算符。

定义 2 $G=(V, E)$是连通图，每一个节点 $v \in V$，每一节点 v 包含一系列接口($i_1^v, i_2^v, \cdots, i_{\deg ree(v)}^v$)。其中，$i_k^v$ 表示节点 v 的第 k 个接口，它的 IP 地址用 $i_k^v.address$ 表示，在 G 中唯一。边 $e \in E$ 通过节点 v_s 的接口 $i_b^{v_s}$ 和节点 v_{s+1} 的接口 $i_a^{v_{s+1}}$ 连接这 2 个相邻节点 v_s 和 v_{s+1}，并且满足关系

$$i_b^{v_s}.address \leftrightarrow i_a^{v_{s+1}}.address \tag{1}$$

定义 3 首选路径 $PP(v_i, v_j) = (V_{PP(v_i,v_j)}, E_{PP(v_i,v_j)})$ 是 G 的一个子图，其中，$V_{PP(v_i,v_j)} = \{v_i, v_{i+1}, \cdots, v_j\}$ 代表从 v_i 到 v_j 的节点序列。

$E_{PP(v_i,v_j)} = \{e_{(v_i,v_{i+1})}, e_{(v_{i+1},v_{i+2})}, \cdots, e_{(v_{j-1},v_j)}\}$ 则代表其间边的序列。一条边 $e_{(v_k,v_{k+1})} \in E_{PP(v_i,v_j)}$ 通过接口 $i_b^{v_k}$ 和 $i_a^{v_{k+1}}$ 连接节点 $v_k \in V_{PP(v_i,v_j)}$ 和节点 $v_{k+1} \in V_{PP(v_i,v_j)}$。依据一些具体应用标准，一条路径可成为首选路径，比如最短路径、费用最低路径等。注意 $PP(v_i, v_i)$并不一定等于 $PP(v_j, v_i)$。

定义 4 $Trace(v_i,v_j)$ 是首选路径 $PP(v_i, v_i)$的函数，调用 traceroute 遍历从 v_i 到 v_j 的每一个节点 $v_k \in V_{PP(v_i,v_j)}$，返回一个接口的列表(每个节点 v_k 只有一个接口)作为输出。

利用已知的 V 中的路径对 $trace(v_i, v_j)$ 和 $trace(v_j, v_i)$，ABR 算法利用式(1)识别 2 条路径间的对称性，定位 $PP(v_i, v_i)$ 和 $PP(v_j, v_i)$之间的点对点连接。ABR 对输入路径对中的 IP 地址进行遍历，返回判定的别名地址对。算法描述如下：

(1) $\overline{V} \leftarrow \varnothing$ (空集)；$\overline{E} \leftarrow \varnothing$；$Alias \leftarrow \varnothing$。

(2)对每一个 $i^{v_k} \in Trace(v_i,v_j) \cup Trace(v_j,v_i)$

1) $\overline{V} \leftarrow \overline{V} \cup V_k$；

2)若存在 $i^{v_{k-1}}$，则 $\overline{E} \leftarrow \overline{E} \cup e(v_k, v_{k-1})$；

3)对每一个 $i_a^{v_s} \in trace(v_i,v_j)$，若存在 $i_b^{v_t} \in trace(v_j,v_i)$，且满足 $i_b^{v_t}.address \leftrightarrow i_a^{v_s}.address$，则 $Alias \leftarrow Alias \cup (i_a^{v_s}, i_b^{v_{t-1}})$。

(3)输出 *Alias*。

ABR 算法首先遍历路径 $trace(v_i,v_j)$ 和 $trace(v_j,v_i)$，计算 $\overline{V}$ 和 $\overline{E}$。之后子图 $\overline{G}$ 包含 v_i 到 v_j 之间的所有连接，通过别名识别尽可能去除冗余的点边信息，缩小 $\overline{G}$ 的大小。在别名识别过程的每一次迭代过程中，算法取 $trace(v_i,v_j)$ 中的一个接口 $i_a^{v_s}$ 与路径 $trace(v_j,v_i)$ 中的接口一一对比，当发现匹配式(1)时，就可以定位一条点对点连接。通过上文的判断，将满足条件的接口标记为 $v_s = v_{t-1}$ 或 $v_t = v_{s-1}$，以此合并 $\overline{E}$ 中相应的边。最后返回所有判定为路由器别名的接口对集合 *Alias*。

4 实验与结果分析

本文使用 C#语言实现 ABR 算法并进行了实验，为更好地构造形如 $trace(v_i,v_j)$ 和 $trace(v_j,v_i)$ 的路径，实验采用多点分布、联合探测的策略，即利用多个探测点对目标地址进行探测，同时探测点与探测点之间互相通信，一个探测点探测得到的路径中包含的新路由器接口地址将交给其他探测点作为目标地址继续探测，不断循环，直到不再出现新的路由器接口地址为止。实验网络为无环网络，路由器之间采用点对点连接，实验网络拓扑如图 7 所示。

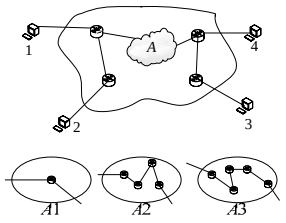

图 7 实验网络拓扑

其中，*A*1, *A*2, *A*3 为区域 *A* 的 3 种情况；1, 2, 3, 4 为 4 个探测点位置，每一次实验探测 8 条路径，分别是：$1 \rightarrow 4$, $4 \rightarrow 1$, $1 \rightarrow 3$, $3 \rightarrow 1$, $2 \rightarrow 4$, $4 \rightarrow 2$, $2 \rightarrow 3$, $3 \rightarrow 2$。与 1 和 4 相连的路由器在实验中保持不变，区域 *A* 为可变区域，按 *A*1, *A*2, *A*3 的结构分别进行 3 次实验，实验从正确性和效率 2 个角度对已有的主动探测算法和本文提出的基于 IP 地址指派惯例的路由器别名识别算法进行比较分析。

4.1 正确性分析

表 2 为各方法对路由器别名的识别情况。

表 2 各方法的识别情况

方法	实际路由器数			探测得到路由器数		
	第 1 组	第 2 组	第 3 组	第 1 组	第 2 组	第 3 组
ABR	5	8	10	5	8	9
Mercator	5	8	10	7	11	15
Ally	5	8	10	6	9	12

可见，ABR 方法比基于源地址和 IP-ID 的探测式算法在识别率上要高 30%~60%。造成差异的主要原因在于：

(1)基于源地址的别名识别算法依赖函数 *GET-SRCADDR* 返回的源地址，而这个返回值受到一些规则的影响，根据文献[8]，探测目标是否为直连网络、路由缓存更新时间、是否配置静态路由都对该方法的正确性产生影响。

(2)网络传播时延和路由器对探测包的处理时延会影响 IP identifier 字段的计数，同时定期更新的路由信息也会改变 IP identifier 的计数。

图 8 比较了随着探测路由器地址数量增加，各方法识别的路由器个数与真实情况的差别，其中，虚线表示“探测数=实际数”的情况。

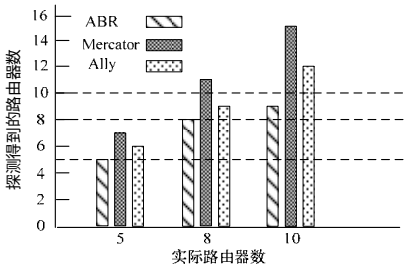

图 8 正确性统计

可以看出，在小型实验网络内，随着路由器数量的增加，探测方式的 2 种方法识别的路由器数量偏离实际情况的幅度明显高于 ABR 方法。

4.2 效率分析

各方法完成别名识别的时间比较如图 9 所示。

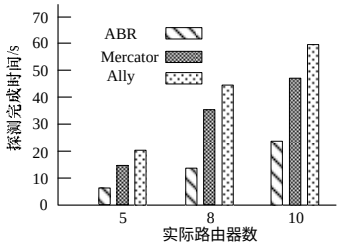

图 9 探测时间比较

可以看出，基于 IP 地址指派惯例的路由器别名识别由于不用发送探测包，只进行对路径的分析，随着路由器数量的增加，其效率明显高于主动探测类算法。对于主动探测算法而言，由于它的计算复杂性为 $O(n^2)$数量级，因此随着路由器接口数的增加，将导致其计算负荷大幅增加。

5 结束语

路由器别名识别是路由器级网络拓扑探测中的重要方面，对探测探测结果的正确性会产生重要影响。本文提出的基于 IP 地址指派惯例的别名识别算法为路由器别名识别提供了新的思路，对中小型规模的网络探测结果效果较好，不受路由器配置的影响，这些特性是当前传统的主动探测方法所不具备的。该算法的约束条件是所探测网络的设计者在路由器之间点对点链路使用/30 和/31 子网掩码这一分配惯例，下一步工作是将这种方法与主动探测方法相结合，得到更好的改善路由器别名识别效果。

参考文献

[1] Gunes M, Sarac K. Importance of IP Alias Resolution in Sampling Internet Topologies[C]//Proc. of INFOCOM'07. Anchorage, Alaska, USA: IEEE Press, 2007.

[2] Govindan R, Tangmunarunkit H. Heuristics for Internet Map Discovery[C]//Proc. of INFOCOM'00. Tel Aviv, Israel: IEEE Press, 2000.

[3] Sping N, Mahajan R, Wetherall D. Measuring ISP Topologies with Rocketfuel[J]. IEEE/ACM Transactions on Networking, 2004, 12(1): 2-16.

[4] Spring N, Dontcheva M, Rodrig M, et al. How to Resolve IP Aliases[D]. Seattle, Washington, USA: University of Washington, 2004.

[5] Hubgard K, Kosters M, Conrad D, et al. Internet Registry IP Allocation Guidelines[S]. RFC 2050, 1996.

[6] Retana A, White R, Fuller V, et al. Using 31-bit Prefixes on IPv4 Point-to-Point Links[S]. RFC 3021, 2000.

[7] Baker F. Requirements for IP Version 4 Routers[S]. RFC 1812, 1995.

[8] Braden R. Requirements for Internet Hosts-communication Layers[S]. RFC 1122, 1989.

编辑 金胡考

(上接第 116 页)

表 2 准确率和平均误码识别率对比 (%)

测试集	流准确率	字节准确率	平均错误率	K-Means 流准确率	K-Means 字节准确率	K-Means 平均错误率
测试集 1	93.87	92.35	15.01	74.31	80.61	21.15
测试集 2	93.98	91.93	11.63	73.41	80.45	20.39
测试集 3	95.54	91.30	10.09	81.93	77.10	16.92
测试集 4	90.53	89.60	10.21	82.11	85.61	18.16
测试集 5	83.66	85.96	12.32	64.53	67.45	16.89
测试集 6	91.38	90.46	20.07	75.21	77.10	26.49
测试集 7	92.09	91.35	16.89	80.92	85.61	19.18
测试集 8	95.80	92.93	12.33	87.62	82.45	17.41
测试集 9	91.02	93.30	16.32	77.80	77.10	20.96
测试集 10	95.11	95.60	12.83	87.37	81.61	17.00
平均值	92.39	91.48	13.87	78.52	79.51	19.46

从表 2 可以看出，本文提出的方法的准确率比 K-Means 方法的准确率提高了 10%以上，同时平均错误率也较 K-Means 方法低，说明根据本文的方法得到的识别模型具有更强健壮性，因而网络环境的变化对该方法的识别率影响较小，可以适合多种网络环境。此时平均错误率在 13%左右，平均错误率偏高的原因是因为部分测试集中像 SSH 等一些应用的流特征向量较少，少数几个这种应用的流属性特征向量被识别错误就可能产生较大的误识别比例，从而产生统计误差。在人为增加这些应用的流属性特征向量后，平均错误率可以降低到 8%左右。

4 结束语

基于信息熵的流特征识别方法的识别效果优于其他基于聚类算法的识别结果。因为聚类算法大多从整体上考虑 2 个特征向量的相似性，所以存在以下缺陷：一是可能忽略或削弱显著特征的作用，二是没有考虑属性的位置关系。

在网络流量识别技术的研究和应用中，可进一步研究的内容包括：(1)流量识别技术可以从 IP 网络进一步扩展到不同的网络体系结构中。(2)用目前的流量识别方法仍有部分流量无法识别，这部分流量占总流量的比例虽然不大，但可能包含了大量的异常流量，对这部分流量的重点识别分析仍有重要的意义。

参考文献

[1] Subhabrata S, Oliver S, Wang Dongmei. Accurate, Scalable in Network Identification of P2P Traffic Using Application Signatures[C]//Proc. of International World Wide Web Conference. New York, USA: [s. n.], 2004: 512-521.

[2] Moorea A W, Zuev D. Internet Traffic Classification Using Bayesian Analysis Techniques[C]//Proc. of ACM SIGMETRICS'05. Banff, Alberta, Canada: [s. n.], 2005: 50-60.

[3] Yuan Jing, Li Zhu, Yuan Ruixi. Information Entropy Based Clustering Method for Unsupervised Internet Traffic Classification[C]//Proc. of IEEE International Conference on Communications. Beijing, China: [s. n.], 2008: 1588-1592.

[4] Xu Kuai, Zhang Zhili, Bhattacharyya S. Profiling Internet Backbone Traffic: Behavior Models and Applications[C]//Proc. of ACM SIGCOMM'05. Philadelphia, PA, USA: [s. n.], 2005: 169-180.

编辑 索书志