

# 基于自适应二分法的 IP 地址分配方法研究<sup>\*</sup>)

陈涛 王峰 王伟 阎保平 毛伟

(中国科学院计算机网络信息中心中国科学院研究生院 北京 100080)

**摘要** 目前 IPv4 骨干路由表已经超过 20 万条,路由表的快速增长已严重影响当前互联网的路由性能。在 IPv6 商用网络即将大规模部署的前期,仔细研究如何规划分配 IPv6 地址,采用合理的地址分配方法,避免下一代互联网络路由表过快增长并加强地址聚合性显得尤为重要。好的地址分配算法可以有效减少地址碎片,控制各级路由表的增长速度并提高地址空间的利用率。本文提出了自适应二分 IP 地址分配方法。经模拟实验表明,这种方法具有良好的聚合能力,为我国运营商制定合理的 IP 地址分配方案提供了理论参考。

**关键词** IPv6, IP 地址分配, 路由聚合

## A Self-adapted Bisection IP Address Allocation Algorithm

CHEN Tao WAGN Feng WANG Wei YAN Bao-Ping MAO Wei

(Computer Network Information Center, Chinese Academy of Science; Graduate University of the Chinese Academy of Sciences, Beijing 100080)

**Abstract** The size of routing table has been beyond 200,000 and its rapid growth has severe impact on router lookup implementations. The carriers also request to get consecutive IP address block for network management and operation. Good IP address allocation algorithm can enhance route aggregation and restrain the growth of routing table. This paper proposes a modified Bisection Scheme, named self-adapted bisection IP address allocation algorithm, for the carriers of our country. The simulation experiment shows that the method has better route aggregation than current one. The allocation method can be used to deploy commercial IPv6 network for the carriers.

**Keywords** IPv6, IP address allocation, Route aggregation

## 1 研究背景

IPv6 技术标准引入了 128 位长的地址结构,从而有效地解决了当前互联网中 IP 地址匮乏的问题。据估计,在合理规划的前提下,IPv6 地址可以持续使用 60~100 年,今后 2~3 代人将使用 IPv6 网络基础设施提供的信息服务。基于 IP 网

络体系结构的多网络融合也正在积极探索中,IPv6 相关技术将会渗透到人民生活的各个领域。

目前,IPv4 骨干路由表已经超过 20 万条<sup>[1]</sup>。根据中国科技网的统计结果来看,当前骨干路由表中我国路由总条数也已经超过 1 万条(图 1)。

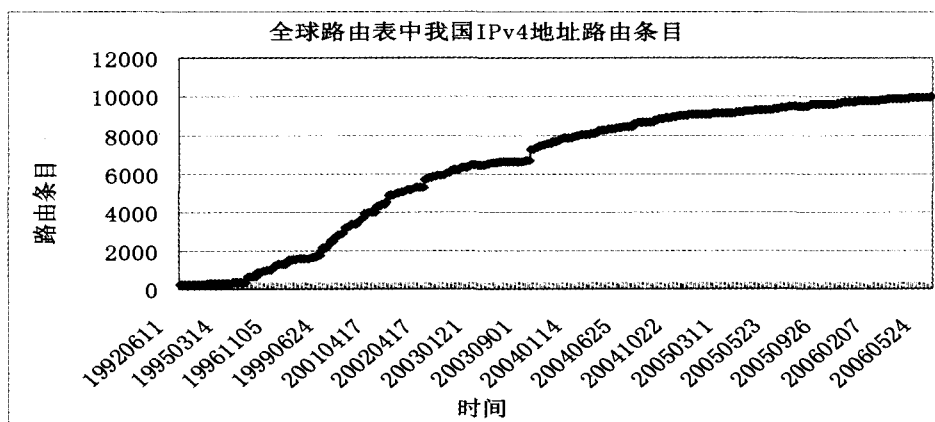


图 1 全球路由表中我国 IPv4 地址路由条目增长表(截至到 2006 年 6 月)

如果地址分配和规划不合理,下一代互联网 IPv6 骨干路由表很可能达到百万量级,不但无法保证电信业务所要求的传输质量,庞大的选路计算负担将超越带宽成为高速互联网

发展的瓶颈。目前我国各大运营商都申请/32 为前缀的 IPv6 地址块进行大规模商用 IPv6 网络部署前的试验工作。如何在实际部署 IPv6 商用网络过程中,根据网络拓扑结构、

<sup>\*</sup>)本课题得到 CNGI 示范工程项目资助(项目编号 CNGI-04-16-2S)。陈涛 博士研究生,研究方向:下一代互联网技术;王峰 博士,研究方向:下一代互联网技术;王伟 博士,研究方向:下一代互联网技术;阎保平 博士生导师,主要研究领域:大规模科学数据共享技术、数据网格技术与应用、下一代互联网技术。

网络规模、网络应用、商业环境等因素,制定易于聚合、便于管理、利用率高的 IPv6 网络地址分配计划,目前还没有太多的分配实践可供参考。

最近的研究表明,好的 IP 地址分配方法对有效减少地址碎片,控制 BGP 路由表的快速增长,提高数据包转发的速度有重要作用<sup>[2,3]</sup>,有利于保证网络快速传输。因此,在我国商用 IPv6 网络部署的早期,就应该仔细研究地址管理和规划、分配方法,为实际的 IPv6 商用网络部署提供参考。

本文第 2 部分介绍地址分配的基本知识,第 3 部分阐述当前 IP 地址分配方法现状及存在的问题,第 4 部分介绍自适应二分 IP 地址分配算法的基本原理及其聚合试验结果,最后对全文进行总结。

## 2 地址分配基本知识

互联网的 IP 地址和 AS 号码分配是分级进行的。ICANN(The Internet Corporation for Assigned Names and Numbers)授权 IANA(The Internet Assigned Numbers Authority)负责全球互联网上的 IP 地址分配工作。IANA 将部分 IP 地址和 AS 号码分配给地区级的互联网注册机构 RIR,然后由这些 RIR 负责该地区的 IP 地址和 AS 号码地址分配、登记注册。通常,RIR 会将地址进一步分配给本地注册机构(LIR)或较大的网络服务提供商(ISP),然后由它们向终端用户做进一步分配。IP 地址和 AS 号码分配的分级示意图如图 2。

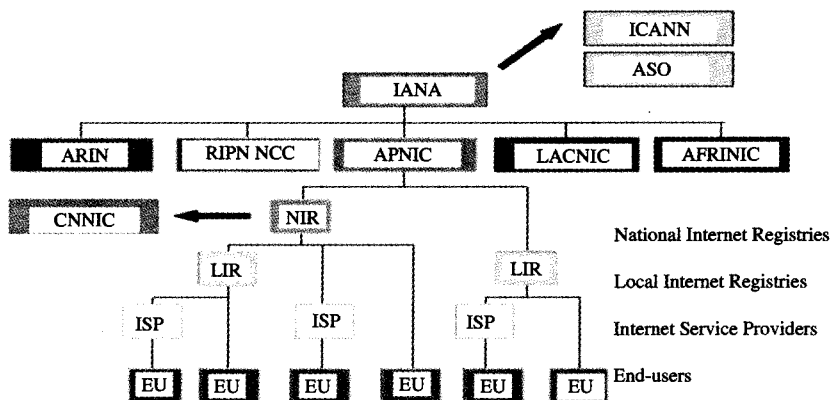


图 2 IP 地址和 AS 号码分配的层次示意图

目前,全球共有 5 个 RIR,分别是:APNIC(亚太地区)、ARIN(北美地区)、LACNIC(南美及加勒比海地区)、RIPE NCC(欧洲、中亚地区和中东地区)和 AFRNIC(非洲)。在 APNIC 管辖的亚太地区,由于其文化和语言的复杂性,设立了国家级注册机构(NIR),负责所在国家/地区的 ISP 和企事业单位的地址分配和管理。

IP 地址分级分配管理的一个主要目标就是在合理的聚合及保留策略的指导下,限制各级路由表快速增长,最大程度减少地址分配的碎片并相应增加地址利用效率。目前,地址分配的基本原则是以 ISP 的独立网络为分配的单元进行分配,这样可以在 ISP 的边界路由器进行地址的聚合,在一定程度上减小地址碎片的数量。然而,ISP 的网络规模的生长是不确定的,在地址租借和不可囤积的政策限制下<sup>[4]</sup>,ISP 间隔性地多次申请,得到的地址空间是不连续的,必然产生一定数量的地址碎片。另外,ISP 网络之间复杂的连接也给出口聚合带来一定的困难,这个问题在下一代 IPv6 互联网中将更为严重。由于目前正处在 IPv6 地址应用的早期阶段,关于 IPv6 地址如何分配的讨论也日趋活跃,众多的分配方法在地址聚合方面显示了不同的特性。

## 3 当前 IP 地址分配方法现状及存在的问题

### 3.1 顺序(Sequential)分配方法

顺序分配方法是一种简单的、自左向右的 IP 地址空间分配管理方法。按照地址请求的先后顺序,从可用地址空间中顺序地把相应大小地址块分配给用户。目前,这种地址分配算法被各级地址分配注册机构普遍采用。这些地址注册机构每次分配地址后,会适当保留临近的地址段,满足用户未来网络的增长需求。比如,APNIC 会为新会员初始分配/32 的

IPv6 地址空间,并为该会员保留临近的 7 块/32 的地址来满足会员未来申请使用,会员的网络可以在/29 的范围内进行聚合广播。

该方法具有直观、简单的分配特性,但是难以容纳连续多次申请的地址需求及突发的大块地址申请。固定的分配保留策略没有根据实际网络规模进行调整的可能,地址利用率也较低。如果中国未来需要的地址量是/10,以/29 作为聚合单元,骨干路由表将有 50 万条,远远超过目前中国 IPv4 骨干路由表 1 万条的数量。

### 3.2 稀疏(Sparse)分配方法

稀疏分配方法又称二分法(Bisection Scheme),最早由 Geoff Huston 在 2004 年提出<sup>[5]</sup>,主要针对北美、亚洲和欧洲等 RIR 中用于 IP 地址分配管理,以解决地址分配过程中地址分段的问题。依照稀疏分配方法,地址块将依照如下规则进行二等分:

每一个新地址块的分配,都会把已有用户和新用户之间的地址块均衡地划分成两段。这种方法能够为两个用户都留下最大的潜在增长空间。如图 3 所示,前四个用户从总体上划分成均等的四部分地址空间。当第五个用户申请一个新地址块的时候,将被分配在图 3(b)中箭头所示空间中,将该部分地址空间划分为两块均等的空间。而第 6、7、8 个用户则会被顺序地分配到第 2、3、4 个用户之后的相应位置。只有在最大的空闲地址片段被耗尽的时候,才会进一步分配下一层较小的地址空间片段。

稀疏分配方法显示了良好的聚合特性,但是其显著缺陷在于并没有充分利用用户的地址申请和使用信息,倾向于将地址空间均分给申请的用户,对不同规模的用户申请适应性较差,还有很大的改进空间。特别是当存在一些增长速度极

快的用户的情况下,会引起相邻用户间不必要的地址冲突和地址空间分段现象。

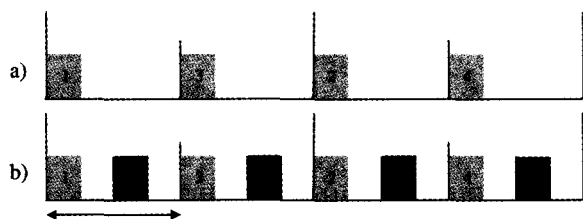


图3 稀疏分配方法

### 3.3 基于增长率的分配方法

基于增长率的地址分配(Growth-Based Address Allocation)方法<sup>[3]</sup>在选择空闲地址块时,要参考已存在用户和新增

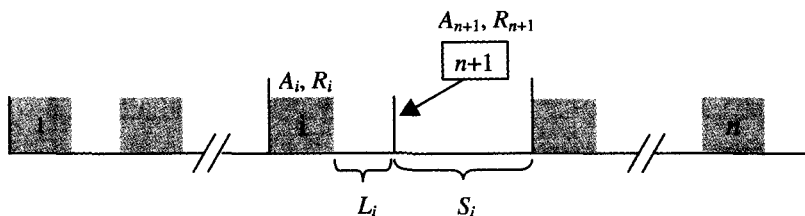


图4 基于增长率的地址分配方法

模拟分配实验表明,基于增长率的方法比现有地址分配方法有更好的聚合效果<sup>[5]</sup>。二分法及基于增长率的方法较适合地区地址分配机构(RIR/NIR),这些区域分配机构有比较充裕的地址池,不需要频繁的地址申请。但是,我国运营商基本没有可供自己分配的地址池,在地址不能囤积的国际分配政策限制下,周期性的申请是必然的结果,所以需要更为符合我国实际情况的分配方法。

## 4 自适应二分 IP 地址分配算法的基本原理及其聚合试验结果

由于我国运营商没有可供分配的 IPv6 地址池,通常是周期性地取得 IPv6 地址块,再按照客户的需求进行进一步的地址分配。在申请周期内,如果能参考地址使用单位的申请习惯和申请频率、业务发展等多种因素,对申请周期内用户地址需求做一定程度的预测,就可以事先做好保留工作,更有效地避免地址块碰撞,进一步减少地址碎片。对不能预测的用户则采用二分地址的方法,最大可能地避免碰撞。根据以上想法,本文提出自适应二分 IP 地址分配算法,该算法尽量避免在高层分配过程中出现地址分段,避免了机械二分地址空间所产生的地址碎片,同时综合考虑客户分配请求的 IP 地址段长度和地址消耗速度的变化情况。在多个用户同时申请分配地址空间时,重点考虑对于总地址空间影响力较大的用户,首先满足该类客户的分配请求。由于大型 ISP 往往处在地址分配结构的较高层,地址分片的结果将导致其下游的地址碎片显著增加。假设顶级运营商的平均地址分段数为  $M$ ,次级分配层次的平均地址分段数为  $N$ ,底层持续增长的客户的地址分片数量将在  $M \times N$  的量级,所以改进的算法在避免过多地址碎片方面也有贡献,实现了对总体分配效果的提升。

### 4.1 自适应二分 IP 地址分配算法的原理

IP 地址分配是分层的,不同层次的地址分配有不同的特点。高层地址申请单位通常是运营商的大区级或省级单位,每年的 IP 地址使用数量可以通过一些方法进行预测和

地址使用单位的地址需求的增长率。当存在  $n$  个已分配地址的用户的时候,对于第  $n+1$  客户最多有  $n$  个可能的地址选择位置。不同于稀疏分配方法对待每一个用户,基于增长率的方法先将这  $n$  个可能的空闲地址块试分给新地址用户,计算出每个试分方案的地址增长空间碰撞时间,选择碰撞时间最长的空闲地址分配方案作为新用户的地址分配方案,如图 4。其计算公式如下:

$$\max\{\min[t(X_i, r_i), t(Y_i, r_{n+1})], i=1, \dots, n\} \quad (1)$$

$t(X, r)$  是地址增长率为  $r$  的地址分配方案  $A$  填满地址空间  $X$  的时间。换句话说,新的分配位置将从所有可用地址空间中选择能够使得当前用户和新用户所导致的地址耗尽时间最长的位置。根据各自增长率  $r$  的不同,函数  $t(X, r)$  也可以表现为其他不同的形式。

并且大区级或省级单位的数量通常情况下是固定不变的。这一层次的地址分配主要目标是在准确预测的基础上,避免分配给各大区的地址块发生碰撞,从而导致在高层地址块分片对底层 ISP 造成重要影响。底层中,小 ISP 业务受市场环境的影响非常大,业务突增或减缓是比较常见的。应该预留足够的地址空间,以便实现更好的地址聚合。对其地址消耗速度变化情况进行周期评价,可以及时释放对于地址需求少的用户所保留的地址段,提高已有地址空间的利用率,以免影响后继申请。

总之,这种地址分配方法能够用于多种层次的地址分配需求,并且在兼顾保证地址空间的聚合性的同时,充分提高地址空间的使用率,能够适用于各种不同大小的地址分配请求。

当客户的申请总量超过当前地址空间中每个空闲地址块的容量,并且运营商的可用地址空间大于当前用户请求量,同时地址空间 HD 比率未达到 0.94 的水平时,自适应 IP 地址分配方法则会按照 2 的幂次对客户的地址申请进行分解,按照分解块从大到小的顺序进行空闲块的最佳适配,从而解决地址利用率和聚合的矛盾。该算法形式化描述如下,设:

- $n$ : 当前已存在的用户数量;
- $n+1$ : 新用户;
- $T$ : 该层地址申请周期
- $A_i; i=1, \dots, n, n+1$ , 用户的地址分配;
- $C_i; i=1, \dots, n, n+1$ , 客户的地址消耗平均速度;
- $S_i; i=1, \dots, n, n+1$ , 空闲地址段  $i$  可分配二分点;
- $M_i; i=1, \dots, n$ , 地址分配位置到可分配二分点的空闲空间;
- $N_i; i=1, \dots, n$ , 用于新分配用户增长的可分配二分点以后的空闲空间;

其主要分配方法和步骤如下:

先选取各空闲段的可分配二分点,

设  $S_i, n$  为空闲地址段  $i$  的中点

if( $S_i, n < A_i + T * C_i$ )

重新设定  $S_i, n+1 = S_i, n$  与  $A_{i+1}$  的二分点, 直到  $S_i, n+1 > A_i + T * C$  (2)  
即可分配二分点不在客户  $i$  的保留区

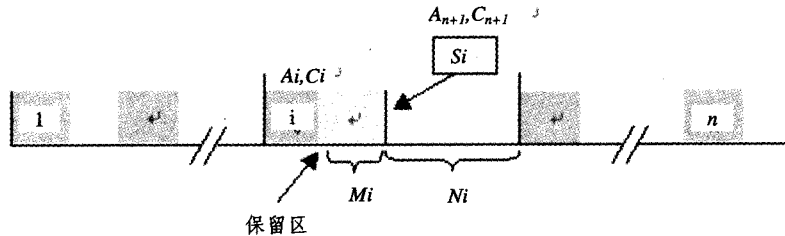


图5 自适应二分 IP 地址分配方法

如果可分配空闲地址块可以满足用户的申请, 则在该位置进行地址空间的划分, 满足用户的此次地址分配请求。

而当每个可分配的空闲地址块均不能满足用户请求量, 但总量仍可以满足该地址分配请求时, 则采取如下方法进行申请拆分, 即将  $A_{n+1}$  进行 2 的幂次分解:

$$A_{n+1} = \sum 2^n \quad (3)$$

将分解出的各分量按照从大到小的顺序进行插入到空闲块中的操作。所选择的空闲块仍然遵循自适应二分地址分配方法的公式(1)、(2)和公式(3), 只不过此时拆分后的地址块消耗的增长速度设为 0。每次较大地址块拆分适配成功后, 剩余地址申请作为新申请执行新的插入过程即可。

为了更为准确地反映每个用户的地址消耗速度的变化情况, 自适应二分法每隔  $T/2$  或  $T/4$  的时间周期, 需要对各个客户的地址消耗速度进行重新评估, 用最新的数值取代原来的数值, 以便使地址分配处理更符合其 IP 地址实际情况。具体可用如下线性经验公式对地址消耗速度进行调整:

$$\text{地址消耗速度} = \frac{\text{最近一次 IP 地址申请量}}{(\text{当前的时间点}) - (\text{最近一次申请时间})} \quad (4)$$

在自适应二分地址分配方法中, 对于大于地址空间中任何一个空闲地址块的 IP 地址分配请求进行 2 的幂次分拆, 可以减少采用地址分拆方法分配带来的地址分片消极影响。因此, 这种自适应二分地址分配方法具有适应不同层次地址分配需求的特点, 降低地址空间中的地址分段数量, 并且有利于充分利用地址空间中的可用空闲地址空间。

#### 4.2 自适应二分 IP 地址分配算法聚合试验结果

为了验证自适应二分 IP 地址分配算法的实际聚合效果, 本文截取在 2002 年至 2004 年期间分配的一段/13 的 IP 地址段进行模拟分配实验, 并与原分配结果进行比较。

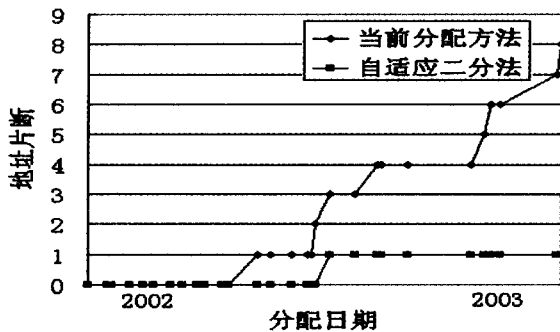


图6 自适应二分 IP 地址分配方法与当前分配方法模拟分配的对比

在这段时间, 国际互联网络地址分配普遍采用顺序分配方法。按照地址分配请求的时间先后顺序, 在地址池空间中选取适当大小的可用地址空间, 分配给相应的地址请求用户。

根据前面公式(1)找到可用地址空间中最晚发生碰撞的位置, 如图 5 中所示。

期间也进行了一定程度的地址预留工作, 但是用统一的自适应二分 IP 地址分配方法进行重新模拟分配后, 地址分片数量明显减少, 图 6 是分片数量统计的比较图。可以看出, 在相同大小的地址分配请求的情况下, 地址空间中所产生的地址片段数量由原来的 8 个降低到 1 个。

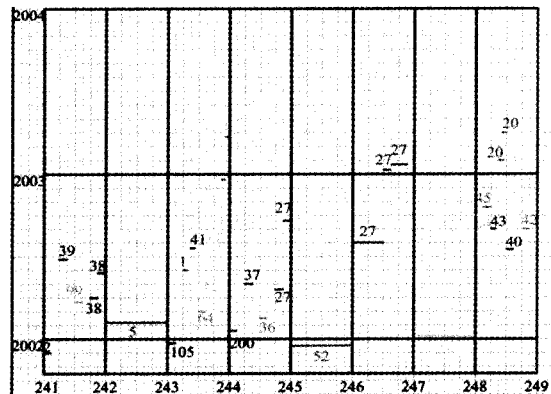


图7 采用当前顺序分配方法的分配结果

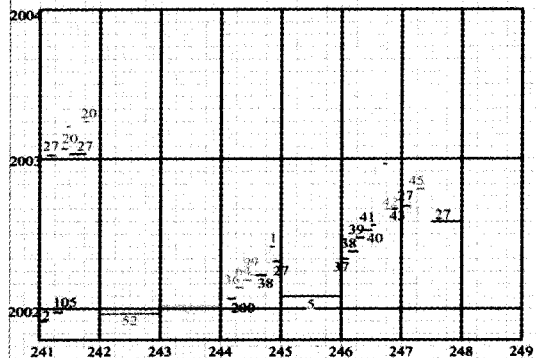


图8 采用自适应二分 IP 地址分配方法的分配结果

图 7 和图 8 是实际的分配结果和模拟分配结果的地址分配图。图 7 中不同颜色的彩色线段和数字代表不同的成员用户, 横坐标是所分配的 8 个 B 类地址, 线段的起止位置以及线段长度代表用户成员所分配地址的大小和地址池空间中的位置, 而纵坐标是地址段所分配的具体时间。图 7 中共有 22 个成员用户的 31 个地址分配请求, 共产生 8 个地址片段。而与之相对的, 在图 8 中可以看到采用自适应二分 IP 地址分配方法后, 只有 1 个地址片段产生。

对比两图可以看出, 在采用自适应二分 IP 地址分配方法后, 以成员用户 27 为例, 其分配的地址空间由原来的 5 块连续地址, 变为 2 块大的连续地址块。同样, 其他成员用户 11、20、38 的地址空间中地址片段都得以避免。事实上, IPv6 地

(下转第 89 页)

类,从而决定是否有违反安全策略的入侵行为发生。

### 3 实验分析

为了测试推进贝叶斯分类法的入侵检测引擎的检测率,我们设计了推进贝叶斯分类法的入侵检测系统(简称为BBIDS),与直接使用贝叶斯分类法的入侵检测系统作比较实验,并进行实验结果分析。

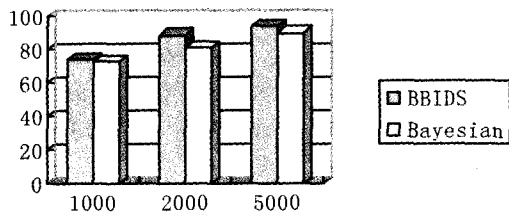


图4 比较实验结果

实验数据选用 KDD Cup 1999 网络数据集<sup>[3]</sup>,为了方便实验我们从中选取了 10000 条数据,这些数据中包含 100 条异常数据。从实验数据中抽取 4 组数据组成 3 个训练子集( $T_1, T_2, T_3$ )和一个测试集( $E_1$ ),3 组训练子集分别包括 1000 个、2000 个和 5000 个带标记数据,测试集包含 1000 个不带

标记的数据。实验中分别使用  $T_1, T_2, T_3$  训练基于引导聚集 ID3 分类法的入侵检测系统和 ID3 分类法的入侵检测系统,然后使用训练好的分类模型对  $E_1$  进行对比测试,得到 3 个测试结果。如图 4 所示。

通过对图 4 的观察,可知训练数据越丰富时,引擎检测检测率越高,BBIDS 的检测率一直比贝叶斯高,当训练数据为 5000 时,BBIDS 的检测率达到了 94.07%。因此,使用推进技术可以提高分类的正确率。使用 BBIDS 的入侵检测系统与使用贝叶斯的入侵检测系统相比具有更高的检测率。

**结论** 推进贝叶斯分类算法的学习需要搜集大量的网络访问的数据进行学习,借此来训练入侵检测系统的学习模型。这是一项非常复杂的工作,但是从实验结果看来,运用加权引导聚集,通过加权学习,可以提高检测率、减小误判,因此该方法运用到入侵检测中取得了良好的效果。

### 参考文献

- 1 Lee W. A data mining framework for constructing features and models for intrusion detection systems; [dissertation of Doctor of Philosophy]. Columbia University, 1999
- 2 Han Jiawei, Kamber M. Data mining: concepts and techniques [M]. San Francisco: Morgan Kaufmann Publishers, 2001. 185~219
- 3 <http://kdd.ics.uci.edu>
- 4 唐正军. 网络入侵检测系统的设计与实现. 电子工业出版社, 2002

(上接第 61 页)

址利用率的衡量标准并不是通常意义的百分比利用率,而是用 HD 比率来衡量<sup>[6]</sup>,其定义为

$$\text{HD 比率} = \frac{\text{Log(已经分配的地址空间)}}{\text{Log(最大可分配地址空间)}} \quad (5)$$

其中,“地址空间”表示/48 的客户数量。HD 比率等于 0.94,相应换算成各前缀的实际使用百分率。可以看出,实际 IPv6 使用率一般不会高于 50%,在宽松的后继申请利用率的条件下,自适应二分法按照 2 的幂次分解进行地址申请的机会小得多,可以预计其聚合性能将会更好。

**结束语** IP 地址分配方法是影响各级路由表增长速度的重要因素,地址使用单位从方便管理和控制运营成本的角度也寻求最大限度的聚类所拥有的 IP 地址段。然而,我国运营商在现行地址分配政策下,只能周期性地申请地址,满足网络扩容的需求,迫切需要一种有效的地址分配方法来指导其进

行 IP 地址分配。模拟实验表明,本文所提出的改进的二分地址分配方法具有良好的聚类特性,可以有效地减少地址碎片并提高地址利用率,为不同层次的地址分配机构实际分配 IPv6 地址提供了有益的参考。

### 参考文献

- 1 IPv4 路由表情况. <http://bgp.potaroo.net/>
- 2 Xu Z, Meng X, Zhang L, et al. Impact of IPv4 Address Allocation Practice on BGP Routing Table Growth. IEEE Computer Communications Workshop (CCW), Oct. 2003
- 3 Wang Mei. A Growth-based Address Allocation Scheme for IPv6 Networking. LNCS, 2005, 3462, 671~683
- 4 APNIC. Policies for IPv4 address space management in the Asia Pacific region. December 2005
- 5 Huston G. Consideration of the IPv6 Allocation Unit Size. <http://www.potaroo.net/drafts/draft-huston-ip6-allocation-unit-00.txt>, June 2005
- 6 Durand A. RFC 3194, The H-Density Ratio for Address Assignment Efficiency: An Update on the H ratio

(上接第 79 页)

点乘运算,而新方案的每个用户要进行  $N$  次  $G_1$  上的点乘运算和  $N$  次公钥加密运算,并且广播的数据量也扩大了  $N$  倍。但这是为原方案的 DC-Net 增加匿名可撤销特性所付出的代价,并且这些操作只需在初始化阶段进行一次。如果采用群签名法实现可撤销匿名性<sup>[3,4]</sup>,则需对每个匿名消息都进行群签名和签名验证,并要增加复杂的零知识证明协议,与本文的方法相比,其额外开销要大得多。

**结束语** 对一种新型的 DC-Net 匿名通信方案进行了改进,改进方案用很低的代价实现了可撤销的发送者匿名性,使得在至少  $t$  个权威执行成员的参与下可以追踪系统中任意匿名消息的发送者。可以证明,改进后的匿名性基于双线性 Diffie-Hellman 判定问题的困难性,在安全性上与原方案是相当的。

### 参考文献

- 1 Chaum D. The dining cryptographers problem; unconditional sender and recipient untraceability. Journal of Cryptology, 1988, 1(1): 65~75

- 2 Chaum D. Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM, 1981, 24(2): 84~88
- 3 Stefan K, Rolf W, Hannes F. Revocable anonymity. In: Proceedings of ETRICS 2006. Heidelberg; Springer-Verlag, 2006. 206~220
- 4 von Ahn L, Bortz A, Hopper N. Selectively traceable anonymity. In: Proceedings of PET2006. Cambridge, UK; Springer-Verlag, 2006. 586~615
- 5 Golle P, Juels A. Dining cryptographers revisited. In: Advances in Cryptology: Eurocrypt' 2004. Berlin; Springer-Verlag, 2004. 456~473
- 6 Waidner M. Unconditional sender and recipient untraceability in spite of active attacks. In: Advances in Cryptology: Eurocrypt' 89. Berlin; Springer-Verlag, 1989. 302~319
- 7 Boneh D, Franklin M. Identity based encryption from the Weil Pairing. SIAM J of Computing, 2003, 32(3): 586~615
- 8 Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612~613
- 9 Cramer R, Damgaard I, Schoenmakers B. Proofs of partial knowledge and simplified design of witness hiding protocols. In: Advances in Cryptology: Crypto' 94. Berlin; Springer-Verlag, 1994. 174~187
- 10 Bellare M, Rogaway P. Random oracles are practical; a paradigm for designing efficient protocols. In: Proceedings of ACM CCS' 93. New York; ACM, 1993. 62~73