软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(7):1908−1923 [doi: 10.3724/SP.J.1001.2012.04125]

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

一种防范 BGP 地址前缀劫持的源认证方案*

刘志辉+, 孙 斌, 谷利泽, 杨义先

(北京邮电大学 信息安全中心,北京 100876)

Origin Authentication Scheme Against BGP Address Prefix Hijacking

LIU Zhi-Hui+, SUN Bin, GU Li-Ze, YANG Yi-Xian

(Information Security Center, Beijing University of Posts and Telecommunications, Beijing 100876, China)

+ Corresponding author: E-mail: kevin2296@gmail.com

Liu ZH, Sun B, Gu LZ, Yang YX. Origin authentication scheme against BGP address prefix hijacking. *Journal of Software*, 2012,23(7):1908−1923 (in Chinese). http://www.jos.org.cn/1000-9825/4125.htm

Abstract: A new origin authentication scheme based on a threaded balanced binary stored hash tree for authenticated delegation/assignment dictionaries is proposed to solve the problems of BGP (border gateway protocol) address prefix hijacking. BGP address prefix announcement is made up of AS number and IP address prefix, and this paper makes use of the number value range to uniformly define two kinds of BGP address prefix announcement resources, so the two kinds of BGP address prefix announcement resources' origin trustworthy problems are issued by one efficient origin authentication scheme in this paper. This scheme inherits the merit of a threaded binary stored hash tree to correct the shortcomings existing in the William Aiello and Patrick McDaniel's origin authentication scheme that the amount of the evidence for invalid delegation/assignment is double that of the valid. Meanwhile, in contrast with original OA scheme, this scheme reduces the number of tree nodes to half the amount of the delegation/assignment attestation set, which is smaller, so this scheme is more efficient.

Key words: origin authentication; prefix hijacking; BGP (border gateway protocol); AS number; IP address prefix

摘 要: 提出了一种基于线索平衡二叉排序哈希树认证委分字典的安全高效的源认证(origin authentication,简称OA)方案,用于防范 BGP 地址前缀劫持攻击.基于 Aiello 和 McDaniel 等人提出的 OA 服务,通过数值区间对 AS 号和 IP 地址前缀这两种 BGP 前缀宣告资源进行了统一的形式化定义,采用一种方案同时解决了两种前缀宣告资源的源可信问题.该方案不仅解决了原 OA 服务中存在的"无效分配关系的证据量是有效分配关系证据量的两倍"的问题,而且与原 OA 服务相比,该方案建树所需要的总节点数降低约一半.同时,委分证据集合的平均长度更小.因此,该源认证方案效率更高.

关键词: 源认证;前缀劫持;BGP(border gateway protocol);AS 号;IP 地址前缀

中图法分类号: TP309 **文献标识码**: A

互联网的域间路由系统是一个大型的分布式系统,在域间路由系统中的各个路由选择域之间运行的协议

* 基金项目: 国家自然科学基金(61003285); 国家重点基础研究发展计划(973)(2007CB310704); 教育部科学技术研究重点项目
收稿时间: 2011-04-29; 定稿时间: 2011-08-24

称为域间路由协议.BGP(border gateway protocol)协议是事实上的互联网域间路由协议标准,当前的最新版本是BGP-4[1].自治系统(autonomous system,简称 AS)是 BGP 中的路由选择域,AS 之间通过 BGP UPDATE 消息交换路由信息,从而实现网络互联.每个 AS 都会向它的邻居 AS 宣告 IP 地址前缀,同时还会将它从某个邻居 AS 那里学到的前缀宣告传播给其他的邻居 AS.然而,并非所有的 AS 都是诚实可信的,也并非所有的 AS 管理者都能保证手工配置正确,缺乏安全机制的 BGP 给这些人留下了安全漏洞,从而可以很容易地实施难以防范的地址前缀劫持攻击[2−9].

攻击者进行地址前缀劫持攻击要么是为了劫持网络流量,要么是为了窃听网络流量[3,10].劫持网络流量主要表现为不实际转发目的地址位于劫持前缀之中的 IP 数据包;而窃听网络流量主要表现为转发目的地址位于劫持前缀之中的IP数据包,实施中间人攻击.由此可见,地址前缀的劫持攻击不仅危及被劫持网络的连通性和安全性,而且可能给整个互联网带来严重影响.

回顾 2008 年 2 月 24 日发生的 AS17557 前缀劫持事件[4−6].巴基斯坦电信管理局(AS17557)出于对国内用户屏蔽 YouTube 的访问目的,非法宣告了 YouTube(AS36561)的网络地址前缀 208.65.153.0/24.其本意是该地址前缀只在本国范围内进行宣告,然后,由于配置错误,使得该地址前缀不仅向国内的提供商宣告,而且还向国外的提供商 PCCW Global(AS 3491)进行了宣告.PCCW Global 收到该地址前缀宣告的时候没有采取任何验证措施,直接将其转发给其他邻居,导致该非法地址前缀宣告在全球范围内蔓延,使得很大一部分用户对 YouTube 的访问被重定向到巴基斯坦电信管理局(AS17557),从而造成这些用户对 YouTube 的访问中断.

由此可知,地址前缀劫持攻击带来的影响和危害已经严重影响了 Internet 的正常运行.因此,我们有必要投入大量的时间和精力来研究检测和防范地址前缀劫持攻击的方法和手段,使得 Internet 向着健康、安全和可信的方向发展.

1 相关研究

近年来,国内外的学者和专家对如何检测和防范地址前缀劫持攻击进行了大量的研究,主要包括:

(1) 基于密码学的攻击避免机制.这类方案根据信任模型又分为集中式攻击避免机制和分布式攻击避免机制:

1) 集中式攻击避免机制.这类机制依赖于严格的层次式 PKI,通过修改 BGP 协议来提供一致的源认证机制,当前的研究主要集中在 S-BGP[11−14],soBGP[15],SPV[16],OA[17,18]和 LAP[19]等.它们都是采用密码技术提供前缀源自治系统的证明,从而在事前避免前缀劫持攻击的发生;

2) 分布式攻击避免机制.这类机制没有一致的信任根,需要通过一些原则选择出信任的对等实体,同样采用密码技术提供前缀源自治系统的证明来积极防范前缀攻击.这类方案的典型代表是psBGP[20].

这两种机制我们统称为“源自治系统认证机制”,简称“源认证”;

(2) 基于检测的攻击发现应急响应机制.这类机制保证当前缀劫持攻击发生时,尽快地检测出攻击,并通知源端采取应急措施.当前的研究主要集中在 pgBGP[21],PHAS[22],Listen&Whisper[23],IRR[24],MOAS List[25]和 E-IRR[10]等.这类方案的优势在于无需扩展 BGP 协议,具有很强的实际部署能力,但它们保护 BGP 系统的安全能力较差.

与被动的攻击检测机制相比,源认证需要修改 BGP 路由协议和路由器软件,而且集中式的源认证机制还需要部署覆盖全网的 PKI,实际部署难度相对较大;而且,当局部进行部署时,只能提供有限的安全性.尽管如此,我们还是要将研究重点放在集中式的源认证机制上.这是因为:首先,攻击检测只是一种当攻击发生时检测出攻击的被动解决方案,且需要快速响应机制的配合,才能够达到快速有效阻断攻击的目的,降低攻击影响,因此,它只是针对问题的一种补救措施,而不是一种根本性的解决方案;其次,源认证机制是一种事先预防机制,它可以有效地防范前缀劫持,同时使用密码学提供技术保障,是一种根本的解决方案,因此,对它的研究也将为下一代可信域间路由协议的制定提供技术支持;再次,分布式的源认证机制对于如何选择一个可信的对等实体作为信任

根是相当困难的,目前还没有特别一致且有效的解决方案.因此,本文以集中式的源认证机制为出发点,研究预防前缀劫持的源认证机制,并提出了新的解决方案.

集中式的源认证机制以 S-BGP 最为经典,S-BGP 中采用了两套树状结构的 PKI 体系:一套 PKI 用于发放证明地址前缀的所有权证书,另一套用于发放证明自治系统号的所有权证书和 BGP 路由器的实体证书.这两套 PKI 的证书发放体系平行于 IP 地址前缀和自治系统号的分配体系,因此,两套 PKI 的根节点都是 ICANN[25].这样建立 PKI 结构的好处是可以不考虑新建 PKI 体系之初遇到的"信任"问题,因而能够保证互联网资源的完全合法使用[26],同时也减小了 PKI 的部署难度,提高了实际应用的可行性.除此之外,S-BGP 引入了证据(attestation)这一概念,用来说明资源的分配关系(说明资源的分配者是谁以及资源的分配对象是谁);与此同时,S-BGP 使用了数字签名的技术来保护证据,从而抵御拜占庭攻击(Byzantine attack),达到资源分配关系的可信性和一致性.证据的产生者使用上面 PKI 体系中的私钥对证据进行签名,而证据的接收者使用上面 PKI 系统中的公钥和证书对证据的签名进行验证,以此来保证证据的真实性和完整性.通过对证据及其签名的有效性和合法性进行验证,可以保证前缀宣告中的地址前缀和自治系统号是来自于合法的分配关系中的,以此保证前缀宣告的可信性,从而抵御前缀劫持攻击.

Aiello 和 McDaniel 等人[17,18]专门对证据的构建和证据签名的构造进行了研究,同时也对地址前缀的委托和分配关系图进行了深入的研究,在此基础上提出了源认证(origin authentication,简称 OA)的概念,通过提供 OA 服务来验证地址前缀的分配关系,并通过其签名的有效性来保证地址前缀宣告的可信性.OA 服务并没有使用 S-BGP 定义的平行于 IP 地址前缀和自治系统号的分配体系的两套 PKI 结构体系,而是可以使用覆盖全球 BGP 的任意的 PKI 体系结构,这样既可以减少 PKI 的部署数量,又给 OA 的部署带来了极大的灵活性.OA 服务使用了 2-3 Merkle 哈希树(Hash tree)构造证据及证据签名,方案不仅安全性高、效率高,而且该方案既可以提供有效分配关系的证据,又可以提供无效分配关系的证据,解决了 S-BGP 存在的分配组织可能会将一种资源同时委托分配给多个客户的安全隐患问题.然而,OA 服务存在的缺陷是对于无效分配关系的证据量是有效分配关系证据量的两倍,而且用于 OA 服务的 2-3 树的构建相对来说比较复杂.王尚平等人[27]在证书撤销问题的研究过程中引入了线索二叉排序哈希树来解决上述 2-3 树中存在的证据量两倍的问题,但是线索二叉排序哈希树没有考虑对树的平衡性,因此在最坏情况下,导致树的高度等于树的节点数,这极大地影响了查询的效率.因此,我们结合二者的优点并对它们各自存在的缺点进行改进,在 OA 服务中引入了线索平衡二叉排序哈希树,提出了一种新的安全、高效的源认证方案.

本文针对上面存在的问题进行了以下两个方面的改进:

(1) 本文修改了 OA 服务中对于地址前缀的定义方式,使用数值区间的方式来定义地址前缀,同时将这种定义延伸到自治系统号的定义,使得本文的方案既适用于对地址前缀的分配关系证明,又适用于对自治系统号的分配关系证明.将研究的范围从只研究地址前缀的分配关系延伸到了对地址前缀和自治系统号的两种分配关系的研究;

(2) 本文在 OA 服务中引入了线索平衡二叉排序哈希树,不仅继承了线索二叉排序哈希树的优点,解决了"无效分配关系的证据量是有效分配关系证据量的两倍"的问题,而且也解决了线索二叉排序哈希树存在的极端不平衡问题.

在介绍本文方案之前,我们首先对方案中使用的记号和术语加以定义.

2 记号和术语

下面我们分别来定义自治系统号、IP 地址前缀、BGP 地址前缀宣告、前缀宣告资源、BGP 发言组织和前缀宣告资源委分组织.

2.1 自治系统号

定义 2.1(自治系统号(autonomous system number,简称 ASN)). 自治系统号也称为 AS 号,是一个 16-bit 的

二进制号码(最大的号码是 65 535),现在由 ICANN 分配.令$\mathcal{ASN}$={1,2,3,…,K}为所有自治系统号的集合,其中,$K=2^{16}$.AS 号段用区间[a,b]表示,其中,$0\leqslant a\leqslant b\leqslant K$,它表示由一段连续的 AS 号组成的集合.注意,对于区间[a,a],它表示单一的 AS 号,即 AS a.此外,对于任何一个区间[a,b],都有[a,b]=[a,i]∪[i+1,j]∪[j+1,b],其中,$0\leqslant a\leqslant i\leqslant j\leqslant b\leqslant K$.对于任意的 $0\leqslant a\leqslant c\leqslant d\leqslant b\leqslant K$,都有[$c$,$d$]是[$a$,$b$]的子集,即[$c$,$d$]⊆[$a$,$b$];[$a$,$b$]是[$c$,$d$]的超集,即[$a$,$b$]⊇[$c$,$d$].如果 $a=c$ 和 $d=b$ 不同时成立,那么我们称[c,d]是[a,b]的真子集,即[c,d]⊂[a,b];[a,b]是[c,d]的真超集,即[a,b]⊃[c,d].

2.2 地址前缀

定义 2.2(IP 地址前缀(address prefix)). 简称前缀,以 IPV4 为例,我们用 $a.b.c.d/j$ 来表示,其中,j 是一个介于 0 和ℓ之间的整数,即 $j\in[0,\ \ell]$;$a.b.c.d$ 是一个ℓ-bit 的 IP 地址,这个 IP 地址末尾的$\ell-j$ 个 bit 位被置为 0.与 AS 号一样,IP 地址前缀现在同样由 ICANN 分配.

为了便于讨论,首先来看文献[17,18]中对于 IP 地址前缀的定义.令$\mathcal{IPA}=\{x|x\in\{0,1\}^{\ell}\}$为所有$\ell$-bit 的 IP 地址组成的集合,对于 IPV4 来说ℓ=32,对于 IPV6 来说ℓ=128.前缀用 x/j 来表示,其中,j 是一个介于 0 和ℓ之间的整数,即 $j\in[0,\ \ell]$;而 x 是一个 j-bit 的数字,即 $x\in\{0,1\}^{j}$.可以很容易发现,一个前缀是由对应的 IP 地址组成的集合,即 $x/j=\{x\cdot y|y\in\{0,1\}^{\ell-j}\}$,它表示前面 j 个 bit 位都等于 x 的所有ℓ-bit 的 IP 地址的集合,其中,·表示一元拼接操作符,$x\cdot y$ 表示 y 拼接在 x 之后(显然,w/ℓ在这里表示由单个 IP 地址组成的集合).

为了本文方案的需要,我们对 IP 地址前缀在数值区间上重新定义.综上可知,地址前缀 $x/j=\{x\cdot y|y\in\{0,1\}^{\ell-j}\}$,它表示前面 j 个 bit 位都等于 x 的所有ℓ-bit 的 IP 地址的集合.可以很容易看到,这个地址前缀是由一些连续的ℓ-bit 的 IP 地址组成的.对于每一个前缀 x/j,我们都可以通过式(2-1)的计算方法将其变换为一个 IP 地址段,即

$$x/j=[(x\cdot\underbrace{000...0}_{\ell-j\text{个}0})_2,(x\cdot\underbrace{111...1}_{\ell-j\text{个}1})_2]=[p,\ q] \tag{2-1}$$

其中,p 和 q 是满足 $0\leqslant p\leqslant q\leqslant 2^{\ell}-1$ 的两个整数.

例 1:以 IPV4 的前缀$(0110)_2$/4 为例,通过上式变换有

$(0110)_2/4=[(0110\ 00000000000000000000000000000000)_2,(0110\ 11111111111111111111111111111111)_2]$

$=[1610612736,1879048191]$.

因此,我们定义前缀为一个 IP 地址段[p,q],其中,[p,q]的计算方法见公式(2-1).

同样地,前缀的子集关系和超集关系都可以使用 IP 地址段对应区间的子集关系和超集关系来表示.例如:

- 如果两个前缀有 $x/j\subseteq y/k$,并且 $x/j=[c,d]$,$y/k=[a,b]$,那么就有[c,d]⊆[a,b];
- 如果两个前缀有 $x/j\subset y/k$,并且 $x/j=[c,d]$,$y/k=[a,b]$,那么就有[c,d]⊂[a,b].

2.3 BGP地址前缀宣告

定义 2.3(BGP 地址前缀宣告(BGP address prefix announcement)). 简称前缀宣告,是前缀和 AS 号的二元关系对($a.b.c.d/j$,n),其中,$a.b.c.d/j$ 表示前缀,n 表示 AS 号.前缀宣告是由 AS 的 BGP 发言人向邻居 AS 发送的,它表示地址前缀 $a.b.c.d/j$ 属于序号为 n 的 AS.

定义 2.4. 我们将前缀和 AS 号统称为前缀宣告资源.

2.4 BGP发言组织

定义 2.5(BGP 发言组织(BGP speaking organization)). 这是可以配置进行前缀宣告的组织,例如那些已经从 ICANN 分配到了 AS 号和前缀的组织.此时,这个组织将其所拥有的 AS 号的一部分分配给自己的 AS 使用,同时将其所拥有的前缀的部分子前缀留给自己的客户主机使用,而不是将所有这些前缀宣告资源的所有权委托给其他组织.这时,该组织就可以将自己的某个 AS 进行编号,同时将部分子前缀分配给该 AS 的客户主机使用.配置完成之后,这个 AS 的 BGP 发言人(BGP speaker)就会向邻居 AS 宣告(announce)这个 AS 序号与该前缀的二元关系对,这样这个组织就成为一个 BGP 发言组织.

2.5 前缀宣告资源委分组织

令$\mathcal{S}$为所有的 BGP 发言组织的集合.对于每一个组织 $C\in\mathcal{S}$,令$\mathcal{PAR}(C)$为当前分配给组织 C 的某种前缀宣告资源的集合,当$\mathcal{PAR}(C)=\mathcal{ASN}(C)$为当前分配给组织 C 的所有 AS 号的集合,当$\mathcal{PAR}(C)=\mathcal{PFX}(C)$为当前分配给组织 C 的所有前缀的集合.

定义 2.6. 我们用$\mathcal{O}$表示前缀宣告资源委分组织的集合,则$\mathcal{O}$为$\mathcal{S}$中所有的组织加上 ICANN、其他地区级的 Internet 注册机构(regional Internet registry,简称 RIR)和其他 Internet 服务提供商(Internet service provider,简称 ISP)的集合,所以,$\mathcal{O}$是所有"拥有"这些 AS 号或者前缀并且可能对这种所有权进行再委托的组织的集合.

3 前缀宣告资源的委分

下面我们将讨论前缀宣告资源委分组织对前缀宣告资源的委分.为了简便起见,本文的其他部分将"前缀宣告资源委分组织"简称为"组织".

定义 3.1(前缀宣告资源的委托(delegation)). 这是指一个组织将自己"拥有"的前缀宣告资源转交给其他组织使用的过程,接受组织有权对所得到的这些资源进行进一步的委托.

定义 3.2(前缀宣告资源的分配(assignment)). 这是指一个组织将自己"拥有"的前缀宣告资源分配给自己的一个或多个 AS 使用的过程.注意到,其实分配是一种特殊形式的委托,即资源委托的对象是自己的 AS 或者是为自己服务的 AS,而只是这些 AS 不能对该资源进行再委托.

可见,对于给定的前缀宣告资源,一个组织既可以对其进行委托,又可以对其进行分配.因此,我们将前缀宣告资源的委托/分配统称为前缀宣告资源的委分.

本文方案中,我们定义 AS 号委分的基本单位是 AS 号段,前缀委分的基本单位是 IP 地址段.由于前缀宣告资源中的 AS 号段和 IP 地址段都可以用表述一致的形如$[p,q]$的数值区间来形式化地加以表示,所以我们使用数值区间$[p,q]$对前缀宣告资源进行统一化表示.

为了讨论方便,我们首先使用文献[17,18]中的方法对前缀宣告资源的委分进行形式化定义.

定义 3.3. 前缀宣告资源的委分链可能经过了多个组织.给定一个前缀宣告资源$[p,q]\subseteq\mathcal{PAR}(C)$,组织 C 可能进行一个或者多个下面的委分:

(1) $(C,RT,[p,q],\mathrm{A}/n)$,即 ASSIGNMENT 选项,表示 C 将$[p,q]$分配给自己的 AS.其中,RT 表示资源类型.当 RT=as,表示是 AS 号段的分配,这时 A/n=A,表示将 AS 号段$[p,q]$分配给了自己的 AS 使用;当 RT=pf,表示是前缀的分配,这时 A/n=n,表示将前缀$[p,q]$分配给了自治系统号 n,即 AS n;

(2) $(C,RT,[p,q],C')$,其中,$C'\in\mathcal{O}$,即 C 将$[p,q]$委托给了组织 C';

(3) $(C,RT,[p,q],\mathrm{R})$,即 RESERVED 选项,表示 C 将$[p,q]$预留,这意味着$[p,q]$既不被分配给自己的 AS 使用,又不被委托给其他组织;

(4) $(C,RT,[p,q],\mathrm{U})$,即 UNAUTHENTICATED 选项,表示 C 对$[p,q]$的委分有待完成.这是为了实现增量部署,组织 C 在对$[p,q]$的委分操作需要时间周期的时候,记录的中间状态.当完成对$[p,q]$的委分操作时,就会将该委分选项变成上面的情况(1)或者情况(2)中的某一种.

组织 C 可能采用上面的零个、一个或者多个选项.由上面的四元组组成的集合就是 C 对前缀宣告资源$[p,q]$的委托/分配策略,简称 C 对$[p,q]$的委分策略(注意,C 对$[p,q]$的委分策略可能是一个空集).对于$\mathcal{PAR}(C)$里面的每一个前缀宣告资源,C 都有一个委分策略,而所有这些策略的集合就是 C 的前缀宣告资源委分策略.$\mathcal{O}$中的每一个组织都有一个前缀宣告资源委分策略.

不失一般性,我们规定,如果$(C,RT,[a,b],\mathrm{A}/n)$,$(C,RT,[a,b],C')$,$(C,RT,[a,b],\mathrm{R})$或者$(C,RT,[a,b],\mathrm{U})$出现在 C 的前缀宣告资源委分策略中,那么相应的$(C,RT,[c,d],\mathrm{A}/n)$,$(C,RT,[c,d],C')$,$(C,RT,[c,d],\mathrm{R})$或者$(C,RT,[c,d],\mathrm{U})$就不允许出现在 C 的前缀宣告资源委分策略中,其中,$[c,d]$是$[a,b]$的任何真子集,这样做可以消除前缀宣告资源委分的冗余性.

3.1 前缀宣告资源委分图

图 1 是前缀宣告资源委分的示意图.

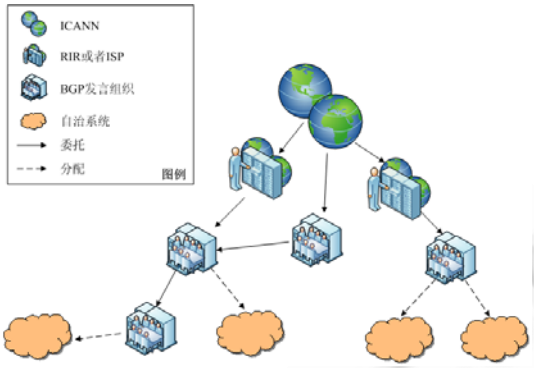


Fig.1 Delegation/Assignment of BGP address prefix announcement resources

图 1 BGP 前缀宣告资源的委分

如图 1 所示,所有前缀委分资源(AS 号和地址前缀)都归 ICANN 所有,ICANN 可以将这种所有权委托给其他组织,而这些被委托的组织可以进一步委托这种所有权.前缀委分资源的最终使用者是 AS,而一个 AS 归属于某个组织,受该组织的管理和控制,因此,AS 的归属组织可以将它拥有的 AS 号和地址前缀分配给该 AS 使用,或者该 AS 进行服务的客户组织也可以将其所拥有的 AS 号和地址前缀分配给该 AS 使用(本质上还是组织之间的委托).

综上可知,AS 号的委分和前缀的委分对于委托部分的定义基本相同,而唯一的区别就在于分配部分.AS 号的分配对象是本组织,所有被分配的 AS 号就归本组织的 AS 使用;而前缀的分配对象是本组织所分配的 AS 号.因此,虽然 AS 号的委分和前缀的委分可能经过不同的组织链路,但是对于委分链路的讨论是一致的.

定义 3.4(委托/分配图(delegation/assignment graph)). 简称委分图,是一个具有边标记的有向图 $G=(V,E)$,它的顶点集和边集定义如下:

- 顶点集 V 是组织集合$\mathcal{O}$和分配对象集合$\mathcal{A}$的并集(对于 AS 号的委分图,分配对象集合就是单点集合{A},即$\mathcal{A}$={A};对于地址前缀委分图,分配对象集合就是 AS 序号集合$\mathcal{ASN}$,即$\mathcal{A}=\mathcal{ASN}$) 除此之外,还有 3 个特殊的顶点,即 R(表示 RESERVED 选项)、顶点 U(表示 UNAUTHENTICATED 选项)和⊥;
- 委分图的有向边表示每个组织的委分策略,对于每一个 C 和每一个 C 的委分策略中形式为$(C,RT,[p,q],Z)$的四元组,都有一条标记了$[p,q]$的有向边,起点是 C,终点是 Z,其中,$Z\in\mathcal{O}\cup\mathcal{A}\cup\{R,U\}$.除此之外,如果 C 对于$[p,q]$的委分策略是一个空集,那么就有一条标记了$[p,q]$的有向边,起点是 C,终点是⊥.

3.2 委分路径的有效性

定义 3.5(委分路径的有效性(validity of delegation/assignment path)). 委分图中的一条有向路径对于$[p,q]$是有效的委分路径,如果满足:

(1) 归属源是 ICANN;

(2) 这条路径对于超集关系是单调的;

(3) 这条路径是无环的;

(4) 分配边的标记是$[x,y]$并且包含$[p,q]$,即满足$[p,q]\subseteq[x,y]$;而且分配边是关系 AS 的.

定义 3.6. 一条委托路径(delegation path),即路径的终点是$\mathcal{O}$中的节点的路径是有效的,只要归属源是 ICANN,并且这条路径是单调无环的.

定义 3.7(空分配(null assignments)). 一条对于$[p,q]$的有效委分路径可能包含一条从 C 到⊥的分配边,这条边表示 C 对$[p,q]$的委分策略是一个空集,我们称其为空分配.

3.3 委分路径的可信性

到目前为止给出的定义还没有消除下面的情况:一个根为 ICANN 的有向树委分图,它的每条路径对于[*p*,*q*]都是一条有效的委分路径.为了看到这一点,考虑下面的简单情形.例如,有一条有效的委托路径终止于 *C*,假设 *C* 已经收到了这条有效委托路径的证据.现在假定 *C* 的委分策略是{(*C*,*RT*,[*p*,*q*],*C*′),(*C*,*RT*,[*p*,*q*],*C*″)},其中,*C*′和 *C*″都不是上面那条委托路径上的节点.从一条终止于 *C* 的有效委托路径,我们可以得到两条有效的委托路径,一条终止于 *C*′,另一条终止于 *C*″.此外,我们将在下面看到,*C* 可能会构建这条终止于 *C*′的委托路径的有效性证据并且发给 *C*′,同时还会构建这条终止于 *C*″的委托路径的有效性证据并且发给 *C*″.

因此,一条委分路径的有效性证据不足以保证 BGP 宣告中地址前缀和 AS 号的二元关系对是唯一的,或者不足以保证在委托路径上的组织没有进行恶意攻击或者操作失误.为了解决这个问题,我们需要更多的定义.

定义 3.8. *C* 的委分策略对于[*p*,*q*]是可信的(faithful),只要 *C* 的委分策略中最多出现下面 4 种情形中的 1 种:

(1) (*C*,*RT*,[*x*,*y*],A/*n*),其中,[*p*,*q*]⊆[*x*,*y*],当 *RT*=as 时,A/*n*=A;当 *RT*=pf 时,A/*n*=*n*,且 $n\in\mathcal{AS}\mathcal{N}(C)$;

(2) (*C*,*RT*,[*u*,*v*],*C*′),其中,$C'\in\mathcal{O}$;

(3) (*C*,*RT*,[*u*,*v*],R);

(4) (*C*,*RT*,[*u*,*v*],U),

其中,[*u*,*v*]是[*x*,*y*]的超集.

定义 3.9(委分路径的可信性(faithfulness of delegation/assignment path)). 委分图中的一条路径对于[*p*,*q*]是可信的,那么只要该委分路径上的每个组织的委分策略对于[*p*,*q*]都是可信的,并且 UNAUTHENTICATED 的委分选项在整个委分路径上最多出现 1 次,而且出现的位置只能是在有效委分路径上的最后一个组织的委分策略中,或者是倒数第 2 个组织的委分策略中.

出现在最后一个组织的委分策略中的 UNAUTHENTICATED 委分选项将来会转化成定义 3.3 中的第(1)种情况;出现在倒数第 2 个组织的委分策略中的 UNAUTHENTICATED 委分选项将来会转化成定义 3.3 中的第(2)种情况.

3.4 委分路径的验证

定理 3.1. 在委分图中,对于前缀宣告资源[*p*,*q*],最多有一条路径既是有效的又是可信的.

因此,对于前缀宣告的接收者来说,下面的信息对于前缀宣告的验证来说是足够的.接收者可以验证:

(1) 委分路径的有效性;

(2) 委分路径上各个组织的委分策略的可信性.

由于 AS 号和地址前缀都可以通过数值区间的方式表示,所以我们可以看到,对于 AS 号 *n* 可以通过数值区间[*n*,*n*]来表示;对于前缀 *a*.*b*.*c*.*d*/*j*,可以通过数值区间[*p*,*q*]来表示.综上所述,我们可以得到如下结论:

定理 3.2. 一个前缀宣告的接收者收到某个前缀宣告(*a*.*b*.*c*.*d*/*j*,*n*),可以通过验证以下信息的有效性来验证这个前缀宣告的有效性:

(1) 对于 AS 号段[*n*,*n*]的 AS 号段委分路径的有效性:

1) 归属源是 ICANN;

2) 这条路径对于超集关系是单调的;

3) 这条路径是无环的;

4) 分配边的标记是[*c*,*d*]并且包含序号 *n*,既满足[*n*,*n*]⊆[*c*,*d*];而且分配边是关系 AS 的.

(2) 该 AS 号段委分路径上各个组织对 AS 号段[*n*,*n*]的委分策略的可信性:

只要 *C* 的委分策略中最多出现下面 4 种情形中的 1 种:

1) (*C*,as,[*c*,*d*],A),其中,*n*∈[*c*,*d*],即[*n*,*n*]⊆[*c*,*d*];

2) (*C*,as,[*a*,*b*],*C*′),其中,$C'\in\mathcal{O}$;

3) (*C*,as,[*a*,*b*],R);

4) $(C,\text{as},[a,b],\text{U})$,

其中,$[a,b]$是$[c,d]$的超集.并且,UNAUTHENTICATED 的委分选项在整个委分路径上最多出现 1 次,而且出现的位置只能是在有效委分路径上的最后一个组织的委分策略中,或者是倒数第 2 个组织的委分策略中.

(3) 对于前缀 $a.b.c.d/j=[p,q]$的前缀委分路径的有效性:

1) 归属源是 ICANN;

2) 这条路径对于超集关系是单调的;

3) 这条路径是无环的;

4) 分配边的标记是$[x,y]$并且包含$[p,q]$,即满足$[p,q]\subseteq[x,y]$;而且分配边是关系 AS 的.

(4) 前缀委分路径上各个组织对前缀 $a.b.c.d/j=[p,q]$的委分策略的可信性:

只要 C 的委分策略中最多出现下面 4 种情形中的 1 种:

1) $(C,\text{pf},[x,y],n)$,其中,$[p,q]\subseteq[x,y]$,$n\in\mathcal{ASN}(C)$;

2) $(C,\text{pf},[u,v],C')$,其中,$C'\in\mathcal{O}$;

3) $(C,\text{pf},[u,v],\text{R})$;

4) $(C,\text{pf},[u,v],\text{U})$,

其中,$[u,v]$是$[x,y]$的超集.并且,UNAUTHENTICATED 的委分选项在整个委分路径上最多出现一次,而且出现的位置只能是在有效委分路径上的最后一个组织的委分策略中,或者是倒数第 2 个组织的委分策略中.

4 前缀宣告资源的源认证

在本文的方案中,我们使用文献[17,18]中定义的源认证标签(origin authentication tag,简称 OAT)这个术语来完成前缀宣告的认证.与文献[17,18]不同,我们对 OAT 进行了扩展,增加了 AS 号的委分证据,从而在一个 OAT 中可以完成两种前缀宣告资源的源认证.OAT 可以与前缀宣告粘附在一起,以带内方式进行宣告;或者也可以由源宣告的接收方通过带外方式获取;或者 OAT 的一部分通过带内方式获取,而另外一部分通过带外方式获取.这完全可以根据实际的应用进行调整.

一个 OAT 包含一条 AS 号委分路径、一个 AS 号委分证据集合、一条前缀委分路径和一个前缀委分证据集合,其中,委分路径上的每一条边在对应的委分证据集合中都有一个元素与之对应.同时,委分证据集合中的每个证据都包含了对证据信息的签名,通过数字签名的技术保证了证据的真实性和完整性.

为了使得一个 OAT 能够得到正确的证明,委分证据集合里的每一个委分证据都必须能够被正确地证明,同时,委分路径的有效性也必须能够被正确地证明.为了验证委托路径的有效性,我们只需简单地验证归属源是不是 ICANN、路径是不是单调且无环的并且分配边是不是关系 AS 的即可完成,这个证明相对来说比较容易完成.而委分证据集合里的每一个委分证据对应了某个组织的委分策略的可信性的证明,这是本文方案所要解决的主要问题.

4.1 委分函数

在描述具体方案之前,我们首先定义组织的委分函数.为了满足委分函数映射的唯一性,我们假定组织的委分策略都是可信的.在后续章节,我们会详细讨论验证委分策略可信性的方法.令$\mathcal{D}(C)$为满足 C 对$\forall[p,q]\in\mathcal{PAR}(C)$都有非空委分策略的所有前缀宣告资源组成的集合.

定义 4.1. 因为我们假定组织的委分策略都是可信的,因此组织 C 的委分策略等同于一个函数 F_C,我们称其为组织 C 的委分函数.它的定义域是$\mathcal{D}(C)$,值域是$\mathcal{O}\cup\mathcal{A}\cup\{\text{R},\text{U}\}\cup\{\perp\}$.亦即,对于每一个$[p,q]\in\mathcal{D}(C)$,$C$ 对于$[p,q]$的委分策略就是$\{(C,RT,[p,q],F_C([p,q]))\}$.

4.2 基于线索化平衡二叉排序哈希树的认证委分字典

首先,我们假定创建委分证据的组织都有签名私钥,以及绑定其公钥和组织身份信息的证书链,这些证书链以全球 BGP 信任的 CA 为根.注意,这里同样采用文献[17,18]中的方法,即允许前缀宣告资源的委分链和公钥证

书链相互独立.这样做是因为这些组织可能想要委托 AS 号或者地址前缀给其他组织,但是不想担当公钥证书的权威机构.

下面我们给出基于线索平衡二叉排序哈希树的认证委分字典(threaded balanced binary stored Hash tree based authenticated delegtion/assignment dictionaries,简称 TBBSHT-ADAD)的抽象化模型.本文方案仍然使用认证字典模型,下面对认证委分字典使用文献[27]中的方法进行定义.

定义 4.2. 令 U 是一个全集,S_C 是 U 的一个子集,即 $S_C \subseteq U$.令 D_{S_C} 是代表 S_C 的一个数据结构,则 D_{S_C} 提供下面几种运算:

(1) 查询运算 $\langle Query,[p,q]\rangle_{S_C}$,代表用户向组织 C 的目录服务查询是否满足 $[p,q]\subseteq[x,y]$ 且 $[x,y]\in S_C$.其回答为 $\langle Answer,[p,q],a_C\rangle_{S_C}$,其中,$a_C\in\{YES,NO\}$,分别对应于肯定和否定的回答.

若其回答为 $\langle Answer,e,a_C,p_C\rangle_{S_C}$,其中 a_C 同上,且 p_C 是由应答组织 C 签名的一个关于 a_C 的证据,则称这样的查询为成员认证查询.

(2) 插入运算 $\langle Insert,[p,q]\rangle_{S_C}$,其中,$[p,q]\subseteq[u,v]$,$[u,v]\in U\backslash S_C$,则插入 $[p,q]$ 后的数据结构为 $D_{S'_C}$.

这里,$S'_C=S_C\cup\{[p,q]\}$.

(3) 删除运算 $\langle Remove,[p,q]\rangle_{S_C}$,其中,$[p,q]\in S_C$,则删除 $[p,q]$ 后的数据结构为 $D_{S'_C}$,其中,$S'_C=S_C\backslash\{[p,q]\}$.

假设 U 是某种宣告资源的全集,S_C 是组织 C 拥有的数值区间的集合,下面我们分别来构造 TBBSHT-ADAD 的数据结构 D_{S_C}.

首先假设组织 C 拥有的数值区间的集合为 $S_C=\{[m_0,n_0],[m_1,n_1],[m_2,n_2],\ldots,[m_k,n_k]\}$.对于区间 $[m,n]$ 来说,我们称 m 为区间 $[m,n]$ 的左值,称 n 为区间 $[m,n]$ 的右值.下面以区间的左值为关键字、以 $[m_0,n_0]$ 为根节点,对 S_C 构建 TBBSHT-ADAD,使每个节点的数据结构如图 2 所示.

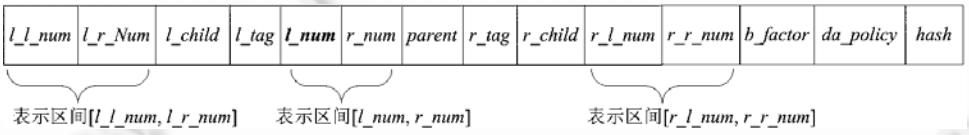


Fig.2 Data structure of TBBSHT-ADAD

图 2 TBBSHT-ADAD 的数据结构示意图

其中,*l_num* 和 *r_num* 分别表示委分区间 [*l_num*,*r_num*] 的左值和右值;*l_child* 和 *r_child* 分别指向其左右子树的根节点;*parent* 指向其父节点;*b_factor* 为节点的平衡因子,用来说明节点的左右子树的平衡情况;*da_policy* 为组织 C 的委分策略(其值是委分函数 $F_C([p,q])$ 的值,即为委托的组织名称或者分配的 AS 号),*l_tag* 为线索左标志,*r_tag* 为线索右标志,*l_l_num* 和 *l_r_num* 分别表示左边未委分的区间 [*l_l_num*,*l_r_num*] 的左值和右值(可能为空(∅)),*r_l_num* 和 *r_r_num* 分别表示右边未委分的区间 [*r_l_num*,*r_r_num*] 的左值和右值(可能为空(∅)),它们的具体含义如下:

(1) *l_tag*=false,表示该节点的左子树不为空且 *l_child* 指向的是该节点的左子树的根,并令 *l_l_num* 和 *l_r_num* 都为空(∅);

(2) *l_tag*=true,表示该节点的左子树为空,令 *l_child* 指向的是该节点的中序前驱节点,并令 [*l_l_num*, *l_r_num*] 为未委分的区间,其中,*l_l_num* 为 *l_child* 指向的节点的 *r_num* 域值加 1,*l_r_num* 为当前节点的 *l_num* 域值减 1.若 *l_l_num*>*l_r_num*,令 *l_l_num* 和 *l_r_num* 都为空(∅),表示 [*l_l_num*,*l_r_num*] 为空(∅),此时,当前节点的 *l_num* 域值为中序前驱节点的 *r_num* 域值加 1(表示两个节点的委分区间相连);

(3) *r_tag*=false,表示该节点的右子树不为空且 *r_child* 指向的是该节点的右子树的根,并令 *r_l_num* 和 *r_r_num* 都为空(∅);

(4) *r_tag*=true,表示该节点的右子树为空,令 *r_child* 指向的是该节点的中序后继节点,并令 [*r_l_num*, *r_r_num*] 为未委分的区间,其中,*r_l_num* 为当前节点的 *r_num* 域值加 1,*r_r_num* 为 *r_child* 指向的结

点的 *l_num* 域值减 1.若 *r_l_num*>*r_r_num*,令 *r_l_num* 和 *r_r_num* 都为空(∅),表示[*r_l_num*,*r_r_num*]为空(∅),此时,当前节点的 *r_num* 域值为中序后继节点的 *l_num* 域值减 1(表示两个节点的委分区间相连).

综上可知,TBBSHT-ADAD 的节点数据结构中使用了 *l_tag*,*l_child*,[*l_l_num*,*l_r_num*],*r_tag*,*r_child* 和[*r_l_num*,*r_r_num*]反映线索信息.当 *l_tag*=false 时,*l_child* 指向其左子树的根;当 *l_tag*=true 时,*l_child* 指向该节点的中序前驱节点,并使用[*l_l_num*,*l_r_num*]记录左边未委分的区间.当 *r_tag*=false 时,*r_child* 指向其右子树的根;当 *r_tag*=true 时,*r_child* 指向该节点的中序后继节点,并使用[*r_l_num*,*r_r_num*]记录右边未委分的区间.

下面我们分别从线索平衡二叉排序树、委分证据的构造和对前缀宣告资源的运算这 3 个角度来分析 TBBSHT-ADAD 数据结构的特性.

(1) 线索平衡二叉排序树

假设组织 *C* 的委分区间的集合为 *S*={[30,50],[80,80],[100,150],[180,200],[220,250],[260,280],[300,360],[400,500],[550,600],[600,700],[700,780],[800,900]},以区间左值作为关键字,构造平衡二叉排序树,如图 3 所示.

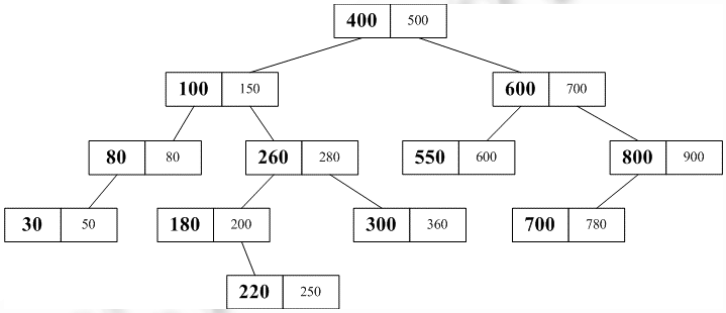


Fig.3 Balanced binary stored tree of TBBSHT-ADAD

图 3 TBBSHT-ADAD 的平衡二叉排序树示意图

利用线索化算法对图 3 所示的平衡二叉排序树进行线索化操作,就可以得到相应的线索平衡二叉排序树,如图 4 所示.

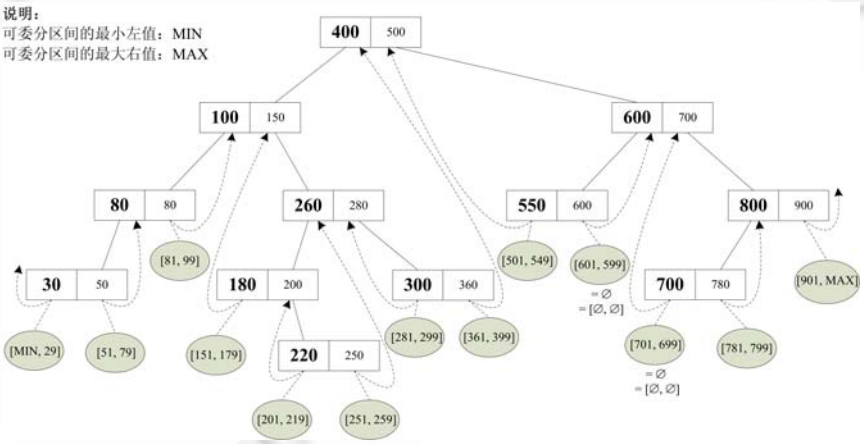


Fig.4 Threaded balanced binary stored tree of TBBSHT-ADAD

图 4 TBBSHT-ADAD 的线索平衡二叉排序树示意图

图 4 中采用虚线箭头表示节点的中序线索化信息.为了直观起见,当节点的 *l_tag*=true 时,左边未委分的区间[*l_l_num*,*l_r_num*]表示为一个左叶子,用虚线连接.同理,当 *r_tag*=true 时,右边未委分的区间[*r_l_num*,*r_r_num*]

表示为一个右叶子,用虚线连接.并且在图 4 中我们规定,当 $n_1>n_2$ 时,区间$[n_1,n_2]$表示为空($\varnothing=[\varnothing,\varnothing]$);当 $n_1=n_2$ 时,区间$[n_1,n_2]$仅为一个点.由此可见,通过线索信息,我们将整个前缀宣告资源区间[MIN,MAX]的资源分配情况在一棵平衡二叉树中得到了充分的呈现.

(2) 委分证据的构造

为了保证上述线索平衡二叉排序树的真实性和完整性,我们利用强抗碰撞 Hash 函数(MD5 或 SHA-1 等)计算每个节点的 Hash 值来填充 TBBSHT-ADAD 数据结构中的 *hash* 域.计算每个节点的 Hash 值可以通过将其左子树根节点 Hash 值(若存在左子树)或该节点的左边未委分的区间[*l_l_num*,*l_r_num*](若不存在左子树),级联该节点的委分区间,再级联该节点右子树根节点 Hash 值(若存在右子树)或该节点的右边未委分的区间[*r_l_num*, *r_r_num*](若不存在右子树),再级联该节点的委分策略,从而得到一个级联的值,对该值进行 Hash 运算的结果值就是这个节点的 Hash 值.

hash 域的值为一个 bit 流字符串(Hash 函数为 MD5 时为 128bit,Hash 函数为 SHA-1 时为 160bit),*hash* 域的计算公式使用公式(4-1)表示.

$$\text{nd}.hash=\text{Hash}(H(PL)\|\text{nd}.l\_num\|\text{nd}.r\_num\|H(PR)\|\text{nd}.da\_policy) \quad (4\text{-}1)$$

公式中的运算符号||表示级联运算,同时,公式中的 $H(PL)$和 $H(PR)$满足:

- 当 nd.*l_tag*=false 时,$H(PL)$=Hash(nd.*l_child.hash*);
- 当 nd.*l_tag*=true 时,$H(PL)$=Hash(nd.*l_l_num*||nd.*l_r_num*);
- 当 nd.*r_tag*=false 时,$H(PR)$=Hash(nd.*r_child.hash*);
- 当 nd.*r_tag*=true 时,$H(PR)$=Hash(nd.*r_l_num*||nd.*r_r_num*).

最后,组织 *C* 只需对根节点的 Hash 值进行签名,即可得到 TBBSHT-ADAD.当然,进行数字签名时要加入时间戳、新鲜性以及有效期,以防止重放攻击.对根节点 Hash 值的签名可以保证对整个数据结构真实性和完整性的认证,任何对 TBBSHT-ADAD 的改动都可被检测出来,除非可以找到 Hash 函数的一个碰撞.

这样得到的 TBBSHT-ADAD 反映了全部前缀宣告资源区间的委分信息,包括委分区间及未委分的区间.对于委分区间和未委分区间的证明可以通过下面的方法来进行:

1) 证明某个区间被委分只需提供区间委分证据 pr_Y:
   (a) 该区间所属的委分区间所在的节点到根节点的路径;
   (b) 路径上所有节点的左、右子树根节点的 Hash 值;
   (c) 组织 *C* 对根节点的签名.
2) 证明某个区间未被委分只需提供区间未委分证据 pr_N:
   (a) 该区间所属的未委分的区间所在的节点到根节点的路径;
   (b) 路径上所有节点的左、右子树根节点的 Hash 值;
   (c) 组织 *C* 对根节点的签名.

查询者在收到证据后可利用 Hash 函数计算得到根节点的 Hash 值,并利用组织 *C* 的公钥验证证据中组织 *C* 对根节点签名的有效性.

(3) 对前缀宣告资源的运算

1) 查询运算 $\langle Query,[p,q]\rangle_{S_C}$

一个成员认证查询 $\langle Query,[p,q]\rangle_{S_C}$ 是查询$[p,q]\subseteq[x,y]$且$[x,y]\in S_C$.目录服务接收到该查询后,按二叉排序树的搜索规则搜索$[x,y]$(使得满足$[p,q]\subseteq[x,y]$):

(a) 若$[x,y]$是一个被委分的区间,目录服务器回答 YES,并给出证据 pr_Y,其中,pr_Y 是由区间$[x,y]$所在的节点到线索平衡二叉排序哈希树的根节点的路径上的所有节点、路径上所有节点的左右子树根节点的 Hash 值和组织 *C* 对根节点 Hash 值的签名构成;

(b) 若$[p,q]$是一个未委分的区间,则$[p,q]$不满足上面条件,但$[p,q]\subseteq[l_l_num,l_r_num]$或者$[p,q]\subseteq[r_l_num,r_r_num]$,此时,目录服务器回答 NO,并且给出证据 pr_N,其中,pr_N 是由该节点到线索平

衡二叉排序哈希树的根节点的路径上所有节点、路径上所有节点的左右子树根节点的 Hash 值以及组织 *C* 对根节点 Hash 值的签名构成.

查询者收到该证据后,对其根节点的 Hash 值进行计算,并利用组织 *C* 的公钥来验证证据中组织 *C* 对根节点 Hash 值的签名,从而可以校验[*p*,*q*]是一个有效的委分区间或者是一个未委分区间.若验证一致,则接受证明;否则,目录服务的回答不可信.目录服务伪造证据相当于目录服务找到了 Hash 函数的一个碰撞,由于采用的是强抗碰撞 Hash 函数,这几乎是不可能的(伪造成功的概率可以忽略).

2) 插入运算$\langle Insert,[p,q]\rangle_{S_C}$

在进行插入运算$\langle Insert,[p,q]\rangle_{S_C}$之前,我们首先需要执行节点查找运算.如果[*p*,*q*]的超集或者[*p*,*q*]的某个子集是一个有效的委分区间,那么终止插入,返回失败.否则,我们可以判断出[*p*,*q*]在插入前为一个无效委分区间,那么它必然包含于某个节点的未委分的区间[*l_l_num*,*l_r_num*](或者[*r_l_num*,*r_r_Num*]).首先查找到该节点,[*p*,*q*]作为该节点的左(或者右)子树的根节点插入,并重新调整二叉排序树使其平衡,然后修改受影响节点的标志、线索化信息及其未委分区间信息和 *hash* 域的数据.

3) 删除运算$\langle Remove,[p,q]\rangle_{S_C}$

同样,在进行删除运算$\langle Remove,[p,q]\rangle_{S_C}$之前,我们首先需要执行节点查找运算.如果[*p*,*q*]不是一个有效的委分区间,那么终止操作,返回失败.否则,我们可以判断出[*p*,*q*]是一个有效的委分区间,假设 nd 指向[*p*,*q*]所在的节点,即待删除节点,则需要分如下 4 种情况进行处理:

(a) 若该节点为叶子节点(即 nd.*l_tag*=true 且 nd.*r_tag*=true),则删除该节点,并重新调整二叉排序树使其平衡,然后修改受影响节点的标志、线索化信息及其未委分区间信息和 *hash* 域的数据;

(b) 若该节点只有左子树,即 nd.*l_tag*=false 且 nd.*r_tag*=true,这时,如果 nd.*r_num*<nd.*parent*.*l_num*,则令 nd.*parent*.*l_child*=nd.*l_child*;如果 nd.*l_num*>nd.*parent*.*r_num*,则令 nd.*parent*.*r_child*=nd.*l_child*.接着删除该节点,并重新调整二叉排序树使其平衡.然后,修改受影响节点的标志、线索化信息及其未委分区间信息和 *hash* 域的数据;

(c) 若该节点只有右子树,即 nd.*l_tag*=true 且 nd.*r_tag*=false,这时,如果 nd.*r_num*<nd.*parent*.*l_num*,则令 nd.*parent*.*l_child*=nd.*r_child*;如果 nd.*l_num*>nd.*parent*.*r_num*,则令 nd.*parent*.*r_child*=nd.*r_child*.接着删除该节点,并重新调整二叉排序树使其平衡.然后,修改受影响节点的标志、线索化信息及其未委分区间信息和 *hash* 域的数据;

(d) 若该节点的左、右子树均非空,即 nd.*l_tag*=false 且 nd.*r_tag*=false,则查找 nd 节点的左子树的最右下节点 rnd,用 rnd 节点取代 nd 节点,并将 rnd.*l_child* 赋给 rnd.*parent*.*r_child*.随后删除该节点,并重新调整二叉排序树使其平衡.然后,修改受影响节点的标志、线索化信息及其未委分区间信息和 *hash* 域的数据.

5 方案讨论

下面,我们对本文提出的 TBBSHT-ADAD 和文献[17,18]提出的 2-3SHT-ADAD 两种方案分别从相同规模节点数的委分证据的平均长度、相同规模节点数的建树时间、相同规模节点数下随机插入一个节点所需要的平均时间、相同规模节点数下随机删除节点所需要的平均时间这几个方面进行对比分析.我们使用 Java 语言编写了测试程序,测试机的配置清单见表 1.

Table 1 Test machine configuration list

表 1 测试机配置清单

操作系统	Windows XP Professional SP3
处理器	Intel(R) Pentium(R) 4 CPU 3.00GHz
内存	2048MB

根据实验测试数据,绘制了图 5~图 11 的对比分析图.其中,图 5~图 8 分析了有效委分区间和未委分区间的

委分证据平均长度的对比,图 9~图 11 分别分析了建树时间、随机插入和删除一个节点所需要的平均时间.

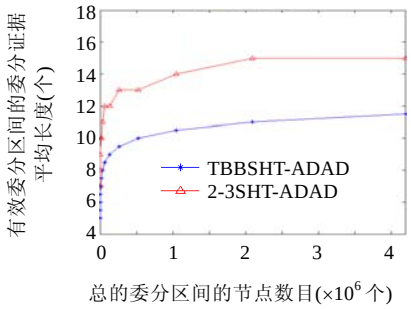


Fig.5 Average length of delegation/assignment attestations for valid number value range on common abscissa

图 5 横轴为普通坐标的有效委分区间的委分证据平均长度

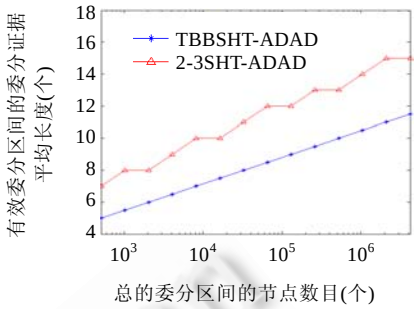


Fig.6 Average length of delegation/assignment attestation for valid number value range on logarithmic abscissa

图 6 横轴为对数坐标的有效委分区间的委分证据平均长度

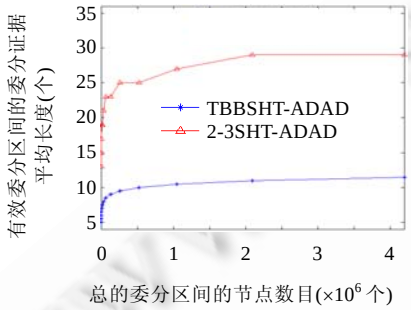


Fig.7 Average length of delegation/assignment attestations for invalid number value range on common abscissa

图 7 横轴为普通坐标的未委分区间的委分证据平均长度

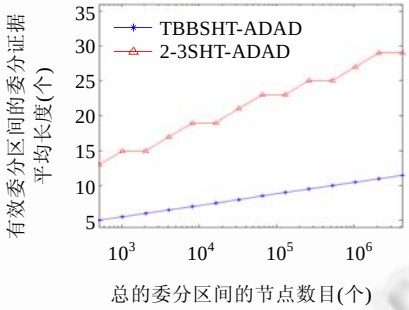


Fig.8 Average length of delegation/assignment attestation for invalid number value range on logarithmic abscissa

图 8 横轴为对数坐标的未委分区间的委分证据平均长度

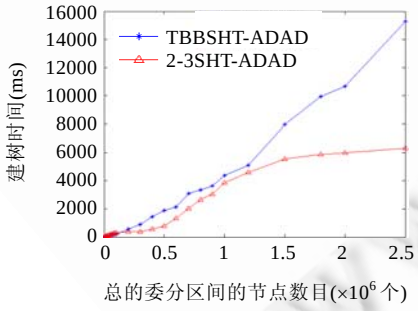


Fig.9 Time of creating tree

图 9 建树时间

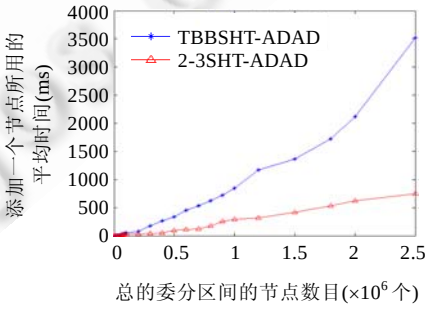


Fig.10 Time of adding a node

图 10 添加节点的时间

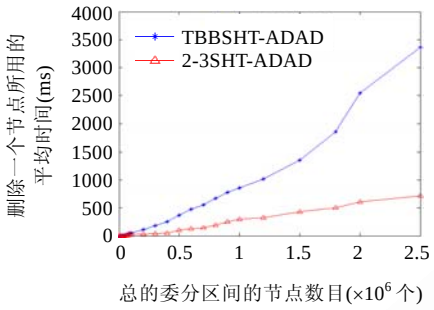


Fig.11 Time of deleting a node

图 11 删除节点的时间

从图 5~图 11 的对比分析中可以发现,与 2-3SHT-ADAD 的源认证方案相比,TBBSHT-ADAD 有以下几个方面的优势:

(1) TBBSHT-ADAD 是基于平衡二叉排序树,相对于 2-3SHT-ADAD 使用的 2-3 树来说,结构简单,不容易出错;

(2) 对于同样多的委分区间,TBBSHT-ADAD 的总节点数等于 2-3SHT-ADAD 的叶子节点树,因此 TBBSHT-ADAD 的总节点数要比 2-3SHT-ADAD 的总节点数少一半左右,从而节省了存储空间;

(3) TBBSHT-ADAD 的委分证据集的长度的平均值比 2-3SHT-ADAD 的委分证据集的长度要短.这是因为,对于同样数量的委分区间,TBBSHT-ADAD 和 2-3SHT-ADAD 的树高基本相同,TBBSHT-ADAD 的委分区间信息及其线索化数据体现在每一个节点上,而 2-3SHT-ADAD 的委分区间信息只存放在叶子节点上,因此,对于某个委分区间的证据,TBBSHT-ADAD 的最短长度是根节点一个节点,最长长度是根节点到叶子节点的路径上的所有节点的个数,即树高;2-3SHT-ADAD 的委分证据长度基本固定不变,等于树高;

(4) TBBSHT-ADAD 对于有效委分区间和未委分区间的证据都只需要一条路径,而 2-3SHT-ADAD 对于有效委分区间的证据是一条路径,对于未委分区间的证据则是两条路径.因此,TBBSHT-ADAD 所需要的证据量更少,需要验证的计算量也更少.

当然,从图 5、图 10 和图 11 的分析来看,TBBSHT-ADAD 比 2-3SHT-ADAD 的建树时间、随机插入和删除节点的平均时间随着节点规模的增长都要快,但是整个域间路由系统的运行瓶颈在路由器上,需要路由器能够快速进行验证.所以,提供更少的证据量和更少的验证量是源认证服务需要解决的真正问题.而对于前缀宣告资源的证据的维护是前缀宣告资源委分组织的工作,它对数据处理时间的要求不是很高,因此牺牲建树时间和节点更新操作的时间换来节点查询和验证的时间的减少,对于域间路由系统的源认证是值得的.

6 结束语

本文对 BGP 前缀宣告进行了研究,完成了对两种 BGP 前缀宣告资源的统一的形式化定义.在此基础上,分析了文献[16,17]提出的 OA 服务存在的“无效分配关系的证据量是有效分配关系证据量的两倍”的缺陷.本文通过提出基于线索平衡二叉排序哈希树的认证委分字典的源认证方案,不仅解决了 OA 服务存在的问题,而且使用一种方案实现了对两种 BGP 前缀宣告资源的源认证.除此之外,与原 OA 服务相比,本文的源认证方案建树所需要的总节点数降低约一半.同时,委分证据集合的平均长度更小.因此,本文的源认证方案效率更高.

References:

[1] Rekhter Y, Li T. A border gateway protocol 4(BGP-4). Internet Engineering Task Force (IETF), RFC 1771, 1995.

[2] Latt KT, Ohara Y, Uda S, Shinoda Y. Analysis of IP prefix hijacking and traffic interception. Int'l Journal of Computer Science and Network Security, 2010,10(7):22−31.

[3] Ballani H, Francis P, Zhang XY. A study of prefix hijacking and interception in the Internet. In: Proc. of the ACM SIGCOMM. 2007. 265−276. http://www.cs.duke.edu/courses/cps214/spring09/papers/hitesh-hijack.pdf [doi: 10.1145/1282427.1282411]

[4] Brown MA, Underwood T, Zmijewski E. The day the youtube died. 2010. http://www.renesys.com/tech/presentations/pdf/nanog43-hijack.pdf

[5] RIPE NCC. YouTube hijacking: A RIPE NCC RIS case study. 2010. http://www.ripe.net/internet-coordination/news/industry-develop ments/youtube-hijacking-a-ripe-ncc-ris-case-study

[6] Dhillon S. YouTube IP hijacking. 2010. http://www.nanog.org/mailinglist/mailarchives/old_archive/2008-02/msg00453.html

[7] Wan T, van Oorschot PC. Analysis of BGP prefix origins during Google's May 2005 outage. In: Proc. of the Security in Systems and Networks. 2006. http://www.ccsl.carleton.ca/paper-archive/twan-ssn-06.pdf

[8] Bono VJ. 7007 explanation and apology. 2010. http://www.merit.edu/mail.archives/nanog/1997-04/msg00444.html

[9] Misel SA. Wow, AS7007!. 2010. http://www.merit.edu/mail.archives/nanog/1997-04/msg00340.html

[10] Liu X, Zhu PD, Peng YX. Internet registry mechanism for preventing prefix hijacks. Journal of Software, 2009,20(3):620−629 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3221.htm [doi: 10.3724/SP.J.1001.2009.03221]

[11] Stephen K, Charles L, Karen S. Secure border gateway protocol (S-BGP). IEEE Journal on Selected Areas in Communication Special Issue on Network Security, 2000,18(4):582−592. [doi: 10.1109/49.839934]

[12] Secure BGP project. 2010. http://www.ir.bbn.com/sbgp/

[13] Stephen K, Charles L, Mikkelson J, Karen S. Secure border gateway protocol (S-BGP)—Real world performance and deployment issues. In: Proc. of the 7th Annual Network and Distributed System Security Symp. (NDSS 2000). 2000. http://www.ir.bbn.com/sbgp/NDSS00/index.html [doi: 10.1109/49.839934]

[14] Steve K. Securing the border gateway protocol: A status update. In: Proc. of the 7th IFIP TC-6 TC-11 Conf. on Communications and Multimedia Security. 2003. 40−53. http://www.net-tech.bbn.com/sbgp/S-BGP_CMS-2003-Kent.pdf [doi: 10.1007/b13863]

[15] Russ W. Securing BGP through secure origin BGP. Internet Protocol Journal, 2003,6(3):15−22.

[16] Hu YC, Perrig A, Sirbu M. SPV: Secure path vector routing for securing BGP. In: Proc. of the ACM SIGCOMM. 2004. 179−192. https://sparrow.ece.cmu.edu/group/pub/hu_perrig_sirbu_spv.pdf [doi: 10.1145/1030194.1015488]

[17] McDaniel P, Aiello W, Butler K, Ioannidis J. Origin authentication in interdomain routing. Computer Networks: the Int'l Journal of Computer and Telecommunications Networking, 2006,50(16):2953−2980. [doi: 10.1016/j.comnet.2005.11.007]

[18] Aiello W, Ioannidis J, McDaniel P. Origin authentication in interdomain routing. In: Proc. of the ACM CCS 2003. Washington, 2003. 165−178. [doi: 10.1145/948109.948133]

[19] Wang N, Zhang JH, Ma HL, Wang BQ. An origin AS verification mechanism based on the length of prefix assignment path for securing BGP. Chinese Journal of Electronics, 2009,37(10):2220−2227 (in Chinese with English abstract).

[20] Wan T, Kranakis E, van Oorschot PC. Pretty secure BGP (psBGP). In: Internet Society Proc. of the Symp. on Network and Distributed Systems Security (NDSS 2005). 2005. http://www.isoc.org/isoc/conferences/ndss/05/proceedings/papers/tao-psBGP.pdf

[21] Karlin J, Forrest S, Rexford J. Pretty good BGP: Improving BGP by cautiously adopting routes. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). 2006. http://www.cs.princeton.edu/~jrex/papers/pgbgp.pdf

[22] Lad M, Massey D, Pei D, Wu YG, Zhang BC, Zhang LX. PHAS: A prefix hijack alert system. In: Proc. of the 15th USENIX Security Symp. Vancouver: USENIX Press, 2006. 153−166. http://irl.cs.ucla.edu/papers/originChange.pdf

[23] Subramanian L, Roth V, Stoica I, Shenker S, Katz RH. Listen and whisper: Security mechanisms for BGP. In: Proc. of the 1st Symp. on Networked Systems Design and Implementation (NSDI 2004). 2004. 127−140. http://www.cs.nyu.edu/~lakshmi/listenwhisper.pdf

[24] Bush R. Validation of received routes. 2010. http://archive.psg.com/001023.nanog/sld001.htm

[25] Internet Corporation for Assigned Names and Numbers. http://www.icann.org

[26] Liu X. Research on security monitoring technologies for inter-domain routing in the Internet [Ph.D. Thesis]. Changsha: National University of Defense Technology, 2008 (in Chinese with English abstract).

[27] Wang SP, Zhang YL, Wang YM. Threaded binary sorted hash trees solution scheme for certificate revocation problem. Journal of Software, 2001,12(9):1343−1350 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/12/1343.htm

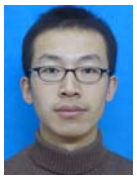
附中文参考文献:

[10] 刘欣,朱培栋,彭宇行.防范前缀劫持的互联网注册机制.软件学报,2009,20(3):620−629. http://www.jos.org.cn/1000-9825/3221.htm [doi: 10.3724/SP.J.1001.2009.03221]

[19] 王娜,张建辉,马海龙,汪斌强.基于前缀分配路径长度的 BGP 源自治系统验证机制.电子学报,2009,37(10):2220−2227.

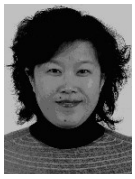
[26] 刘欣.互联网域间路由安全监测技术研究[博士学位论文].长沙:国防科学技术大学,2008.

[27] 王尚平,张亚铃,王育民.证书吊销的线索二叉排序 Hash 树解决方案.软件学报,2001,12(9):1343−1350. http://www.jos.org.cn/1000-9825/12/1343.htm



刘志辉(1982－),男,山西怀仁人,博士生,主要研究领域为信息安全,密码学.

谷利泽(1965－),男,博士,副教授,主要研究领域为数字签名技术与应用.



孙斌(1967－),女,博士,副教授,主要研究领域为计算机网络,网络安全.

杨义先(1961－),男,博士,教授,博士生导师,主要研究领域为密码学,网络安全.