基于数字签名的防污染网络编码方案①

周赵斌, 许 力

(福建师范大学 福建省网络安全与密码技术重点实验室, 福州 350007)

摘 要: 网络编码技术对于提高网络吞吐量、均衡网络负载、提高带宽利用率、增强网络的鲁棒性等方面都有明显的优势, 但是无法直接抵抗污染攻击. 最近, 学者提出了基于同态哈希函数的签名方案, 可以较好检测污染攻击, 但是很难定位被污染的节点. 本文结合两者的优势提出了一个基于数字签名的网络编码方案, 该方案不仅能够抵抗污染攻击, 而且能有效地确定出攻击源的位置, 从而降低污染攻击对网络造成的影响, 并提升网络的健壮性.

关键词: 网络编码; 污染攻击; 身份确认; 数字签名; 同态哈希函数

Pollution-Resistant Network Coding Scheme Based on Digital Signature

ZHOU Zhao-Bin, XU Li

(Fujian Normal University, Fujian Provincial Key Laboratory of Network Security and Cryptology, Fuzhou 350007, China)

Abstract: Network coding technology has obvious advantages for improving networks throughput,balancing network load,improving bandwidth availability ratio, enhancing robustness of networks, but it can't resist against pollution attacks directly. Recently, scholars propose the signature scheme based on homomorphic Hash function, which could better detect pollution attacks, but it is difficult to locate the contaminated node. This paper proposes a network coding scheme based on the digital signature by combining with the advantages of both. It can not only resist pollution attack, but also can effectively identify position of attack source. Thus, it reduces the impact of pollution attacks to the network and enhances the robustness of the network.

Key words: network coding; pollution attacks; identification; digital signature; homomorphic hash function

绪论

2000 年 Ahlswede 等[1]首次提出了网络编码理论, 与传统单一的存储——转发路由功能不同的是: 它允许中间节点在转发数据消息之前对来自不同链路上的消息数据进行编码, 进而提高网络的吞吐量和鲁棒性. 2003 年, Li 等[2]证明了线性网络编码可以使网络的传输容量达到网络多播的最大流, 突破了传统路由传输的"瓶颈"问题. 随后, Ho 等人[3]提出了随机网络编码, 其是在网络中参与传输的节点, 其输出信道上传输的数据是该点多条输入信道上传输的数据的随机线性组合, 证明了通过随机线性网络编码, 能以极大概率达到网络多播的最大流和证明了接收节点能以很大的概率正确恢复出信源所发送的信息.

由于网络编码在提高网络吞吐量、减少时延、改善负载均衡、节省节点能耗和增强网络鲁棒性等方面均显示出其优越性, 所以其可广泛应用于 Ad Hoc 网络、传感器网络、P2P 内容分发和网络安全等领域. 随着网络中的通信量越来越大, 对信息数据进行安全的传输就显得越来越迫切. 当节点进行网络编码时, 其数据信息进行编码操作(如异或), 所以它在很多方面可以达到一定的数据安全要求. 但, 近期很多学者在对其安全性方面的研究发现: 在网络编码的系统中, 虽然它提高了网络的吞吐量和可靠性, 但同时也带来了不可忽视的安全问题, 如污染攻击和拜占庭攻击等. 污染攻击是网络编码安全领域的热点研究问题, 攻击者

① 基金项目:国家自然科学基金海峡联合基金(U1405255);国家自然科学基金(61202450,61472083,61202452)
收稿时间:2015-10-22;收到修改稿时间:2015-11-25 [doi:10.15888/j.cnki.csa.005192]

损坏网络节点并注入新的污染消息形成新的编码数据包. 由于网络编码中的每一个节点不仅具有路由功能而且具有编码功能, 如果某个中间节点被敌手捕获利用, 并向网络中注入恶意的数据包, 由于网络编码是要对信息进行编码混合操作, 那么经过它所编码转发的数据包会进一步地转发到其它的可信节点进行编码混合, 因此攻击者只需要注入少量的恶意数据包就可以达到污染全局的目的. 在网络编码的网络环境下, 此类攻击完全可以导致污染像瘟疫一样在网络中蔓延, 具有很大的破坏性. 传统的签名认证方案被证明不能够抵抗污染攻击, 因为经过网络编码后, 原有的消息签名会因为节点是否参与编码而情况不同, 如果参与编码则签名消息会被破坏, 如果不参与签名, 那么需要在每个传输的消息后面加上签名, 这样会导致消息分组过长且需要解码才能认证消息, 继而会增加时间开销和空间开销. 目前基于密码学的方案在抵抗污染攻击中得到了较好的应用.

密码学领域中的诸如数据加密、哈希函数和消息认证等方式可以确保数据的安全传输. 在编码过程中, 节点通过在数据包增加额外的认证信息, 之后让接收者对数据消息进行消息认证, 这样可以让中间节点识别并过滤掉被污染的报文. 目前常用的密码学方案中主要有同态哈希函数和同型数字签名. Krohn 等[4]提出基于同型哈希函数的机制, 让源节点为每一个原始的数据包计算哈希值, 并通过可靠信道传输到各中间节点. 由于其具有同态的性质, 所以中间节点可以独立地计算出哈希值, 进而达到验证的目的. Kehdi 等[5]主要利用网络编码的零空间来实现认证, 克服了 Krohn 等人方案中计算量大的缺点. Yu 等[6]提出的基于数字签名的方法, 要求为每一个新文件分发新的公共密钥, 并且密钥的大小与文件大小成线性关系, 这一要求限制了其在大容量文件分发系统中的使用. 在文献[7]中, Yu 等又利用对称密码的方法, 采取不同于哈希和数字签名的方法, 但带宽消耗大的问题依旧存在. 但是 Yun 等人[8]已经证明了 Yu 等人的方案不具有同态性质, 其网络安全性很弱. 以上解决方案中[4-8]都是以牺牲系统性能作为代价的, 也就是说为了防御污染攻击而运行的协议都会大幅度降低使用网络编码所带来的性能改善. Zhang 等人结合同态消息认证码和同态签名设计出了一种混合加密方案[9]. Gkantsidis 等人[10]提出了一种基于同态哈希函数的方案, 该方案通过增加一条额外的安全信道用来传输原始消息的哈希值, 实现了中间节点对信源节点发送的消息进行完整性验证功能, 从而判断网络中的消息是否被篡改. Charles 等[11]在文献[10]的基础上提出了一种可用来检测网络数据包是否被篡改的同态数字签名方案, 并且该方案可以对网络中的节点进行身份认证. 文献[12]中, 徐光宪等人基于列表译码法, 设计出一种能同时抵抗强窃听和污染攻击的安全网络编码方案.

本文基于数字签名技术, 提出一种能抵抗污染攻击的网络编码方案. 在我们的方案中, 源节点发送的消息使用其私有密钥加密, 而中间节点使用源节点的公钥验证收到的信息. 这可以使得源节点发送的原始消息不被窃听者窃听, 可有效地防止窃听攻击. 因为现有的网络编码复杂度存在较大的缺点, 从信息分组的角度来分析, 减少网络编码的操作数目是降低编码操作复杂性最理想的方式, 所以文中使用了同态哈希函数功能, 它允许中间节点为他们输出的消息生成的签名很容易由之前输入的签名消息得到, 即输出的消息是由输入的消息生成. 由于每个节点都附加到它的输出签名消息, 其下游节点可以有效地验证收到的消息和丢弃被污染或者伪造的消息. 本方案的优势是比较适用于资源受限的网络, 如无线传感器网络. 在本方案中, 不仅解决了传统传输方式中需增加额外的安全信道问题, 而且能提供源认证批量验证. 最后, 本文将传统公钥签名方案与同态哈希函数相结合, 每个节点发送的签名消息都会附上一个签名验证密钥 *id*, 作为整个数据包的一部分发送到下游节点, 下游节点能够根据对应的验证密钥 *id* 确定出接收到的签名是由哪一个节点发送的, 从而达到确定出污染节点位置信息的目的.

1 预备知识

1.1 网络编码定义

网络编码的本质是利用节点的计算能力提高链路带宽的利用率, 图 1 中 S 是信源, 中间节点有 U、V、W、X(其中 W 是带 b1 和 b2 同时从 S 传到信宿 Y 和 Z. 若采用传统路由方法(如图 1(a)所示), 由图易知, S 与 Y, Z 之间均分别存在两条相互独立的路径. 由于两组路径间存在共有链路 WX, b1, b2 不能同时在链路 WX 上传输, 则 S 到 Y, Z 的最大信息流速率为 1.5 比特 / 单位时间. 若采用图 2(b)的网络编码方法, 在节点 W 上对 b1, b2 执行异或操作并转发, 则节点 Y 可以通过 $b1 \oplus b1 \oplus b2$ 的计算解出 b2, 同理 Z 也可以解出

b1. 因此, 可使 S 到 Y, Z 的信息流速率达到 2 比特 / 单位时间, 带宽利用率提高 33%.

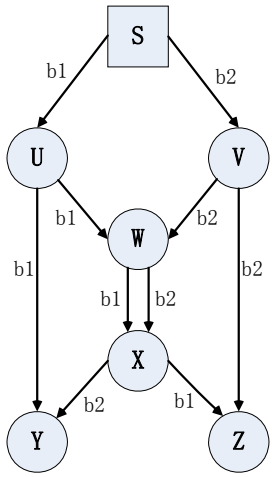

图 1 (a)路由方法

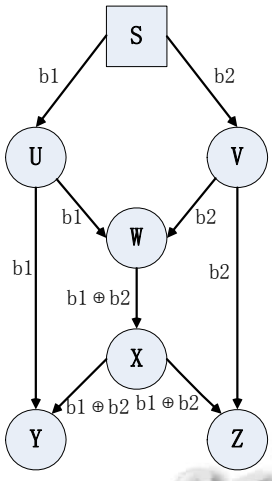

图 2 (b)网络编码

单源和多源是目前主要的两种网络编码模型. 单源模型的发送源节点仅有一个, 多源模型的发送源节点有多个. 本文中我们设定的网络模型是单源多播. 我们用一个有向非循环的图 $G=(V,E)$ 表示, 其中 V 表示网络中节点集合, E 表示网络中边的集合.

1.2 系统模型

在本网络中, 系统模型参考文献[6], 源节点 S 通过中间节点(模型中的节点 1, …, 7)同时发送 n 个消息 M_1, …, M_n 到目的节点 t_1, …, t_k.

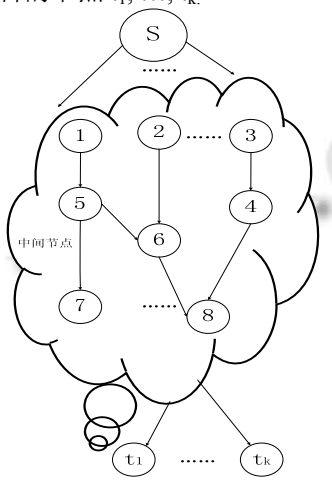

图 3 基于网络编码的多播网络模型

2 基于数字签名的网络编码方案

2.1 随机线性网络编码算法

将源节点发送的消息 F 按顺序分割为 m 个向量 $v_1,v_2,...,v_m\in F_q^n$, $v_i=(v_{i1},v_{i2},...,v_{in})$, F_q 是一个包含 q 元素的有限域, m, n, q 是预先设置的系统参数. 为了原始消息的安全性, 源节点在传播之前首先必须对这些向量 v_i 进行编码, $i=1,2,...,m$.

由于本文利用随机线性网络编码, 所以网络中的每一个中间节点都将对接收到的向量 $(u_1,u_2,...,u_l)$ 进行线性叠加, 并生成编码后的消息 u 转发出去. 因此, u 可表示为该中间节点所收到的消息 $(u_1,u_2,...,u_l)$ 的线性组合, 即

$$u=(\alpha_1,\alpha_2,...,\alpha_l)\times\begin{pmatrix}u_1\\u_2\\\vdots\\u_l\end{pmatrix},$$

其中 $\alpha=(\alpha_1,\alpha_2,...,\alpha_l)$ 为中间节点随机产生的局部编码向量.

为了便于信宿节点对编码消息解码并恢复出原始消息, 必须在每个原始文件向量的前面加上一段长度为 m 的单位向量, 扩充方式如下所示.

$$v_i'=(\overbrace{\underbrace{0,0,...,1}_{i},0,...,0}^{m},v_i)$$

$$=(\overbrace{\underbrace{0,0,...,1}_{i},0,...,0}^{m},v_{i,1},v_{i,2},...,v_{i,n})\in F_q^{m+n},1\le i\le m$$

中间节点对信源发送的消息向量进行线形编码组合, 组合向量为:

$$u'=\sum_{i=1}^{n}\beta_i v_i'$$

$$=(\beta_1,\beta_2,...,\beta_m,\sum_{i=1}^{m}\beta_i v_{i,1},\sum_{i=1}^{m}\beta_i v_{i,2},...,\sum_{i=1}^{m}\beta_i v_{i,n},)$$

$$=(\beta_1,\beta_2,...,\beta_m,u_1,u_2,...,u_n),$$

其中 $\beta=(\beta_1,\beta_2,...,\beta_m)$ 为全局编码向量, 作用是为目的节点提供解码信息, 从而解出原始消息 $(v_1,v_2,...,v_m)$.

为了方便, 消息向量记为:

$$u'=(w_1',w_2',...,w_m',w_{m+1}',w_{m+2}',...,w_{m+n}').$$

由于从信源处收到的 m 条编码消息为线性无关, 所以当目的节点收到 m 条线性无关的消息 $(u_1',u_2',...,u_m')$.

即: $(u_1',u_2',...,u_m')$

$$=\begin{bmatrix} w_{1,1}' & \cdots & w_{1,m}' & w_{1,m+1}' & \cdots & w_{1,m+n}' \\ w_{2,1}' & \cdots & w_{2,m}' & w_{2,m+1}' & \cdots & w_{2,m+n}' \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ w_{m,1}' & \cdots & w_{m,m}' & w_{m,m+1}' & \cdots & w_{m,m+n}' \end{bmatrix}$$ 时,

该目的节点可以解码得到原始消息.

原始消息的恢复过程如下所示. 首先, 令消息向量$(u_1', u_2', ..., u_m')$中的右边 m 列向量构成的矩阵为 P, 左边 n 列向量构成的矩阵为Q, 然后计算 P^{-1}, 解出原始消息:

$$(v_1, v_2, ..., v_m) = P^{-1}Q .$$

2.2 数字签名方案

近年来, 许多基于同态消息认证码的方案被提出来用于抵抗污染攻击, 但是这些方案都面临同样一个问题, 即源节点产生的消息认证码 *MAC t* (MAC, Message Authentication Code)属于很小的有限域 , 这也就意味着只要敌手能够随机的知道 *MAC t* 的确切含义, 也就能够以 的概率成功实施污染攻击, 而通常. 同时, 网络无法确定受到污染攻击的节点位置, 节点只能在收到一个或几个邻居节点发送的数据包消息后, 再编码签名时才会去确定网络中是否存在污染攻击. 若节点发现网络中有污染存在, 此节点并不能定位到是哪个节点发送的消息受到污染(在这里污染节点是被恶意敌手捕获还是出现故障不做讨论). 针对上述两个问题, 本文设计出一个基于同态消息认证码的签名方案, 此方案中, 参与的有网络通信节点和一个可信节点 *TAN*(Trust Authentication Node). 源节点生成发送消息并用私钥签名计算哈希值. 中间节点对接受到的父节点发送的哈希值和签名进行验证, *TAN* 的工作是产生公共参数和确定污染节点位置, 它不仅能够抵抗污染攻击而且能有效地确定出攻击源的位置. 中间节点需要验证上一节点发送的签名消息, 如果节点验证成功, 则需要与 *TAN* 通信并及时更新系统签名参数, 如果验证失败, 则根据从 *TAN* 获取的验证公钥确定受污染攻击节点的身份. 签名方案包括签名算法及节点身份确认两部分.

2.3 签名算法

数字签名在信息安全领域, 包括身份认证、数据完整性验证以及不可否认性等方面发挥着重要的作用. 同态哈希函数其优点在于: (1)能够对数据持有性进行有效地证明; (2)能对数据的完整性进行有效地保护; (3)同态哈希函数计算量小, 可以极大地减少对带宽的需求. 目前大多数的数字签名都是基于大整数分解或离散对数计算的困难性等数学难题, 如 1989 年被提出来的 Schnorr 签名, 由于该签名方案具有计算量小, 速度快等特点, 所以其被广泛应用到网络安全等领域.

本方案中签名算法(Sch NC)过程由参数建立算法、签名算法、验证算法三个算法组成.

2.3.1 参数建立算法

此过程主要由源节点和一个可信的第三方节点 *TAN(Trust Authentication Node)*完成, 首先它们共同协商选择一个大素数

$$p, q \in Z_P^*, P > 2^{1024}, q > 2^{160}, q \mid (p-1);$$

生成一个 g, $g \in Z_P^*$, 是一个本原元, 且满足 $g^q = 1 (\mathrm{mod}\ p)$. 具体算法如下.

(1)源节点

① 选择一个参数$x, 0 < x < q-1, x \in Z_p^*$, 作为源节点签名的私钥. 并计算: $y = g^{-x} mod\ p$;

② 设 $k_i \in Z_{p-1}^*, i = 1, 2, ..., m$, 计算: $R_i = g^{k_i} (\mathrm{mod}\ p)$,

其中每个 k_i 是源节点为每—个件块建立的系统秘密参数. 然后计算:

$$c_i = H(R_i, v_i', y),\ d_i = (k_i + c_i x) (\mathrm{mod}\ q),$$

最后生成消息-签名对: $(v_i', (c_i, d_i))$

(2) 中间节点

① 中间节点随机选择 $x_i \in Z_p^*$, i 表示第 i 个中间节点.

② 计算: $y_i = g^{-x_i} (\mathrm{mod}\ p)$.

③ 设 $k_{id} \in Z_{p-1}^*, i = 1, 2, ..., m$, 先计算:

$R_{id} = g^{k_{id}} (\mathrm{mod}\ p)$, 其中每个 k_{id} 是源节点为每—个中间节点建立的系统秘密参数, i 表示第 i 个中间节点. 再计算:

$$c_{id} = H(R_{id}, u', y_i),$$
$$d_{id} = (k_{id} + c_{id} x_i) (\mathrm{mod}\ q),$$

最后生成消息-签名对: $(u', (c_{id}, d_{id}))$.

④ 中间节点每个都有一个唯一的标识身份 $id \in \{0,1\}^*$. 私钥 $SK = (x, x_i)$) 为签名私钥, $PK = (p, q, g, y_i, H)$ 为公钥. 由 *TAN* 公开参数 $\{PK, H, id, k_i, k_{id}\}$.

2.3.2 签名算法

由于网络中包含了源节点和中间节点, 所以该算

法包括对源节点和中间节点的签名过程, 在这里定义两个签名函数, 如下表 1 所示.

节点	输入	输出
源节点	$sign\ V\ (SK, v_i', k_i)$	$(v_i', (c_i, d_i))$
中间节点	$sign\ U\ (SK, u', k_{id})$	$(u', (c_{id}, d_{id}), id)$

(1) 计算签名: $R_i = g^{k_i} (\mathrm{mod}\ p)$,

$c_i = H(R_i, v_i', y)$, $d_i = (k_i + c_i x) (\mathrm{mod}\ q)$,

源节点输出签名后的消息: $(v_i', (c_i, d_i))$;

(2) 中间节点对接收到的消息做线性编码组合:

$u' = \sum_{i=1}^{m} \beta_i v_i'$, $\beta = (\beta_1, \beta_2, ..., \beta_m)$ 为全局编码向量, 编码后的有效数据包消息为: $(u_1, u_2, ..., u_n)$;

(3) 计算签名: $R_{id} = g^{k_{id}} (\mathrm{mod}\ p)$,

$c_{id} = H(R_{id}, u', y_i)$, $d_{id} = (k_{id} + c_{id} x_i) (\mathrm{mod}\ p)$;

中间节点输出签名后的消息: $(u', (c_{id}, d_{id}), id)$

2.3.3 验证算法

接收节点对收到的消息签名进行验证, 由于已知公钥: $PK = (p, q, g, y_i, H)$, 节点收到消息-签名对: $(u', (c_{id}, d_{id}), id)$, 只需要判断 c_{id} 是否等于 $H(g^{d_{id}} y_i^{c_{id}} (\mathrm{mod}\ p), u', y_i)$.

证明.

$$\begin{aligned} g^{d_{id}} y_i^{c_{id}} (\mathrm{mod}\ p) &= g^{k_{id} + c_{id} x_{id}} y_i^{c_{id}} (\mathrm{mod}\ p) \\ &= g^{k_{id} + c_{id} x_{id}} \left(g^{-x_i}\right)^{c_{id}} (\mathrm{mod}\ p) \\ &= g^{k_{id} + c_{id} x_{id}} g^{-c_{id} x_{id}} (\mathrm{mod}\ p) \\ &= g^{k_{id}} (\mathrm{mod}\ p) \end{aligned}$$

又因为: $R_{id} = g^{k_{id}} (\mathrm{mod}\ p)$

所以: $c_{id} = H(R_{id}, u', y_i)$

证毕.

如果验证通过, 则节点接收签名和消息, 否则拒绝. 由于验证时中间节点接收到的签名是 $(u', (c_{id}, d_{id}), id)$, 里面包含着节点的位置信息, 如果签名不通过, 中间节点可以将验证失败时所解析出的污染节点位置发送到可信的第三方节点 *TAN*, 并由第三方可信节点 *TAN* 删除此污染节点.

3 方案分析

3.1 安全性分析

3.1.1 抵抗污染攻击

为了防止网络中的污染攻击存在, 本文利用了 Schnorr 签名和同态哈希函数的安全性. 首先签名发生在源节点处和中间节点处, 假设源节点是可信任的, 也就是说攻击者不可能在源节点处发起污染攻击. 对于中间节点来说, 攻击者可以捕获网络中的任何节点并利用它发起攻击. 如在此节点发送污染的信息或者伪造签名并发送给下一个节点, 但是他通过 *PK* 计算签名私钥是很困难的, 因为 $y = g^{-x} (\mathrm{mod}\ p)$, 攻击者想通过该等式解出签名所使用的私钥 x 相当于解离散对数问题. 同时因为在中间节点签名的过程中同一个节点的 k_{id} 对每一个接收到的消息进行签名, 产生的消息签名对是不一样的, 所以攻击者也不能够使用 k_{id} 和两个不同的签名来计算私钥 x.

3.1.2 抵抗窃听攻击

由于信源节点对发送的消息进行了扩展和数字签名, 签名的私钥由可信的源节点随机选择, 攻击者获取不到, 所以对窃听攻击者来讲是安全的, 也就是信源节点的签名消息是安全的. 所以即使窃听者能窃听到消息—签名对 $(v_i', (c_i, d_i))$, 敌手也就解不出关于 $(v_1, v_2, ..., v_i)$ 的任何有意义的信息. 从信息论的角度上看, 此方案达到弱安全要求的网络编码.

3.2 效率分析

由于本方案中的运算只涉及到了模指数运算和乘法运算并没有使用线性对运算, 因此根据运算的相似性, 可与文献[6]中 Yu 等人的 *RSA* 签名方案进行比较. 设 T_{me} 为模指数运算所需时间, T_{mul} 为乘法运算所需时间. 表 2 中对比模指数运算, 表 3 中对比乘法运算. 从表 2 和表 3 可以看出, 本文方案在效率上优于 Yu 等人的方案. 但是总体上运算时间基本相同, 但是考虑本方案能在抵御污染攻击的基础上还能确定出污染节点的位置信息, 所以本文还是要优于 Yu 等人的方案. 而且本文考虑到了信源发送的消息的安全问题, 采取了的防范措施, 确保了原始消息的安全, 有效地抵御了窃听攻击.

表 2 模指数运算时间对比

方案	签名	验证
Yu 等人	(m+n+1)T_{me}	(m +2)T_{me}
本文	(m+n)T_{me}	(m +2)T_{me}

表 3 乘法时间运算对比

方案	签名	验证
Yu 等人	(m+n)T_{mul}	(m +2)Tmul
本文	(m+n)T_{mul}	(m +1)Tmul

4 结论与展望

污染攻击是网络编码中一个重要的安全问题, 本文针对这个问题设计出一种基于数字签名的网络编码方案, 它在抵抗污染攻击的同时, 还能有效地确定出攻击源的位置. 本方案除了可抵抗污染攻击, 而且可同时抵御窃听攻击. 但该方案仍存在着带宽消耗较大的问题.

云计算的出现为网络编码带来了一个全新的研究领域. 由于云的计算能力很高, 而网络编码正是利用节点的计算能力来提高网络的传输服务, 为使网络编码在云计算领域中得到广泛的应用, 安全的网络环境是一个重要的前提. 在综合考虑安全效率和网络性能的前提下, 如何选择一个折衷点是我们今后要研究的方向. 还有本文网络编码的应用环境是单源网络, 而现实中许多的网络为多源网络[13]. 由于两者网络存在的差异, 所以使得本文方案中的单源网络编码方案不能直接应用到多源网络中. 牛彩芬等人在文献[14]中, 针对数据完整性问题, 利用同态 Hash 函数和向量合并算法, 提出了一种抗污染攻击的多源线性网络编码数据完整性验证方案.

参考文献

1 Ahlswede R, Cai N, Li SYR, et al. Network information flow. IEEE Trans. on Information Theory, 2000, 46(4): 1204−1216.

2 Li SYR, Yeung RW, Cai N. Linear network coding. IEEE Trans. on Information Theory, 2003, 49(2): 371−381.

3 Ho T, Koetter R, Medard M, et al. The benefits of coding over routing in a randomized setting. Proc. of the IEEE International Symposium on Information Theory. 2003. 442.

4 Krohn MN, Freedman MJ, Mazieres D. On-the-fly verification of rateless erasure codes for efficient content distribution. Proc. IEEE Symposium on Security and Privacy, 2004. IEEE. 2004. 226−240.

5 Kehdi E, Li B. Null keys: Limiting malicious attacks via null space properties of network coding. Proc.of INFOCOM'09. IEEE Press. 2009.

6 Yu Z, Wei YW, Ramkumar B, et al. An efficient signature-based scheme for securing network coding against pollution attacks. Proc. of the 27th IEEE Conference on Computer Communications (INFOCOM). Phoenix, AZ, US. IEEE Press. April 13-18, 2008. 2008. 1409−1417.

7 Yu Z, Wei YW, Ramkumar B, et al. An efficient scheme for securing XOR network coding against pollution attacks. Proc. of IEEE INFOCOM. Rio de Janeiro, Brazil. IEEE Press. April 19-25, 2009. 2009. 406−414.

8 Yun A, Cheon JH, Kim Y. On homomorphic signatures for network coding. IEEE Trans. on Computers, 2010, 59(9): 1295−1296.

9 Zhang P, Jiang YX, Lin C, et al. Padding for orthogonality: Efficient subspace authentication for network coding. Proc. of IEEE INFOCOM. Shanghai, China. IEEE Press. April 10-15, 2011. 2011. 1026−1034.

10 Gkantsidis C, Rodriguez PR. Cooperative Security for Network Coding File Distribution. Proc. of the 25th IEEE International Conference on Computer Communications (INFOCOM). Barcelona, Spain. IEEE Press. April 23-29, 2006. 2006. 1−13.

11 Charles D, Jain K, Lauter K. Signatures for network coding. International Journal of Information and Coding Theory, 2009, 1(1): 3−14.

12 徐光宪,付晓.一种基于列表译码法的改进的安全网络编码.小型微型计算机系统,2013,34(4).

13 张玉洁,蔡英,李卓.网络编码中抗污染攻击研究.北京信息科技大学学报(自然科学版),2013,1:18

14 牛淑芬,王彩芬,张玉磊,等.多源网络编码数据完整性验证方案.计算机工程,2015,3(3):21−25.[doi:10.3969/j.issn.1000-3428.2015.03.004].