文章编号:1001 - 9081(2011)06 - 1512 - 03

doi:10.3724/SP.J.1087.2011.01512

抵御污染攻击的双源网络编码签名算法

牛淑芬,王彩芬,刘雪艳

(西北师范大学 数学与信息科学学院,兰州 730070)

(sfniu76@nwnu.edu.cn)

摘　要:网络编码易遭受污染攻击的破坏,而传统的签名技术不能适用于多源网络编码。基于一种离散对数问题安全的向量哈希函数,提出一种有效抵御污染攻击的双源线性网络编码签名算法,方案中每个源节点用自己的私钥对文件签名,中间节点可用向量的合并算法线性组合来自不同源的消息,且中间(信宿)节点仅用公钥就可验证收到的签名。方案的安全性依赖于 Co-Diffie、Hellman 问题,并在随机预言模型下,证明能够抵抗信源节点和中间节点的攻击。

关键词:双源网络编码;哈希函数;双线性对;编码系数;离散对数

中图分类号: TP309.7; TP393.08　　**文献标志码**:A

Signature scheme for securing two-source network coding against pollution attacks

NIU Shu-fen, WANG Cai-fen, LIU Xue-yan

(*College of Mathematics and Information Science, Northwest Normal University, Lanzhou Gansu* 730070, *China*)

Abstract: Networks coding is highly susceptible to pollution attacks, but such attacks cannot be prevented by the standard technology of signature. Based on the vector hash which is secure if the discrete logarithm problem is infeasible, an efficient signature scheme for securing two-source networks coding against pollution attacks was proposed. In this scheme, each source node signed the files with its own private key, the intermediate nodes, with the merge algorithm, produced linear combinations of vectors from different files. The intermediate nodes could verify the received signature solely by the public key. The security of signature scheme relies on the hardness of the Co-Deffie-Hellman problem. Under the random oracle model, the new scheme is proved to be secure against the source nodes and intermediate nodes attacks.

Key words: two-source network coding; Hash function; bilinear pairing; coding coefficient; discrete logarithm

0　引言

文献[1]提出了网络编码的理论,对于网络编码中的污染问题,传统的数字签名方法都不适用,于是,人们提出一些新的签名算法来适应网络编码的需求。对单信源的网络编码签名算法,有两类签名技术:一类是基于同态哈希函数的签名[2-3],此类技术需要安全信道传输哈希值;另一类是同态数字签名技术[4-5],此类算法采用信源节点的私钥签名,利用签名函数的同态性,中间(信宿)用公钥就可验证数据的真实性。

然而,单源网络编码签名算法无法应用于多源网络编码中,其主要原因在于:现有的单源网络编码签名算法只需用一个私钥来对消息进行签名,而在多源网络编码中,用不同的私钥签名会破坏现有某些算法中签名的同态性;其次,多源网络编码需要线性组合来自不同信源节点的消息(或向量)。文献[6]的多源网络编码签名算法基于 BGLS 的环签名方案,该方案的缺点是每个信源节点要计算一组签名,计算量相对大。文献[7]提出了一个多源网络编码的短签名算法,并在随机预言模型下证明其安全性,该算法的不足在于每个源节点只能传输一个消息向量;并且需要一个可信第三方为其每个源节点生成私钥。文献[8]利用签名函数同态性提出了一个安全的签名算法,该方案每个源节点用自己的私钥签名,中间节点用每个源节点的公钥验证。文献[9]系统地提出了一个针对多源网络编码的数字签名,对于来自不同信源节点(不同子空间)的向量,文献[9]首次提出了向量和向量空间的合并算法,给出了一个多源网络编码的签名体制,并定义了安全性,遗憾的是没有给出具体的签名算法。

本文考虑双源网络编码签名算法,基于文献[9]所提出的向量的合并算法,利用文献[2]提出的同态哈希函数(Vector Hash,VH)对每个源节点的消息签名,基于此哈希的安全性及私钥的安全性,本文提出的算法能抵抗源节点之间的伪造攻击,并用随机预言模型证明了能够抵抗中间节点的伪造攻击。方案的安全性依赖于离散对数困难问题和 Co-Deffie-Hellman 问题。

1　基础知识

1.1　双线性映射和 Co-Deffie-Hellman 问题

首先介绍有关双线性映射和 Co-Deffie-Hellman 问题的一些记号:

1) G_1,G_2 是两个有着相同素数阶 q 的乘法循环群;

2) g_1 是 G_1 的生成元,g_2 是 G_2 的生成元;

3) $\psi: G_2 \to G_1$ 是同构映射,$\psi(g_2) = g_1$;

收稿日期:2010 - 11 - 26;**修回日期**:2011 - 01 - 22。

基金项目:国家自然科学基金资助项目(61063041);甘肃省高等学校研究生导师科研项目(1001-09)。

作者简介:牛淑芬(1976 -),女,甘肃通渭人,讲师,博士研究生,主要研究方向:信息安全;　王彩芬(1963 -),女,河北安国人,教授,博士生导师,主要研究方向:信息安全、电子商务协议;　刘雪艳(1978 -),女,甘肃临洮人,讲师,硕士,主要研究方向:信息安全。

4) $e:G_1 \times G_2 \to G_T$ 是可计算的双线性映射。

1.1.1　Co-CDH 和 Co-DDH[10]

1) 计算 Co-Deffie-Hellman(Co-CDH):给定 $g_2, g_2^a \in G_2$, $h \in G_1$,计算 $h^a \in G_1$

2) 判定 Co-Deffie-Hellman(Co-DDH):给定 $g_2, g_2^a \in G_2$, $h, h^b \in G_1$,若 $a = b$ 输出 YES,否则输出 NO,当输出 YES 时,我们称 (g_2, g_2^a, h, h^b) 是 Co-Deffie-Hellman 组,即:

$a = b \Leftrightarrow e(h, g_2^a) = e(h^b, g_2)$。

1.1.2　双线性映射

设 G_1, G_2, G_T 是有着相同素数阶 q 的乘法循环群,g_1 是 G_1 的生成元,g_2 是 G_2 的生成元,双线性映射 $e:G_1 \times G_2 \to G_T$ 满足下列性质:

1) $e(u^a, v^b) = e(u,v)^{ab}$ 对任意 $a, b \in Z_p, u \in G_1, v \in G_2$;

2) 对任意 $u_1, u_2 \in G_1, v \in G_2, e(u_1 u_2, v) = e(u_1, v) \cdot e(u_2, v)$;

3) 对任意 $u, v \in G_2$, $e(\psi(u), v) = e(\psi(v), u)$。

1.2　双源线性网络编码

1.2.1　基本概念

多源网络编码用一个无环有向图 $G = (E, V)$ 来表示,其中 E 为网络中链路的集合,V 为网络中所有节点的集合。网络中有两个信源节点 $S = (s_1, s_2)$ 需要向网络中的另一个节点集合 $T \subset V$ 中的所有信宿节点传输消息,每个信源节点在传输之前对它要发送的信息进行分块处理,把要发送的文件分成 m 个分块,每个分块用一个 n ($n \gg m$) 维的行向量来表示,行向量的元素为有限域 F_q^n 内的数据。文件可以表示为:$\boldsymbol{V} = (\boldsymbol{V}_1, \boldsymbol{V}_2, \cdots, \boldsymbol{V}_m)$,$\boldsymbol{V}_i$ 表示文件的第 i 个分量,其中 $\boldsymbol{V}_i = (v_{i1}, v_{i2}, \cdots, v_{in})$。设多播网络中任意一条边上发送的消息为 $y_i = (y_{i1}, y_{i2}, \cdots, y_{in})$,中间节点收到 l 条消息的线性组合,即:$\boldsymbol{y} = \sum_{i=1}^{l} \beta_i y_i$,其中 $\boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_l)$ 称为全局编码系数向量,为了使接收节点可以方便解码,编码向量 $\boldsymbol{\beta}$ 可以附在消息后面可以和消息一起传输。此时扩展的原消息可以看做向量空间 F_q^{m+m} 中的一个向量,记为:

$$\boldsymbol{V}_i = (\hat{v}_i, 0) = (v_{i1}, v_{i2}, \cdots, v_{in}, \overbrace{0, \cdots 0, 1, 0, \cdots 0}^{m}) = (v_{i1}, v_{i2}, \cdots, v_{in}, v_{in+1}, \cdots, v_{im+n}) \subset F_q^{m+n}$$

编码消息可以表示为:

$$\boldsymbol{y} = (y_1, y_2, \cdots, y_n, y_{n+1}, \cdots, y_{m+n}) \subset F_q^{m+n}$$

1.2.2　向量的合并算法[9]

假设 s_1 源点发送的消息为:

$\boldsymbol{V} = (\boldsymbol{V}_1, \boldsymbol{V}_2, \cdots, \boldsymbol{V}_m)$

s_2 源点发送的消息为:

$\boldsymbol{W} = (\boldsymbol{W}_1, \boldsymbol{W}_2, \cdots, \boldsymbol{W}_m)$

对于任意 $\boldsymbol{V}_i, \boldsymbol{W}_i \in F_P^{m+m}$, $\boldsymbol{V}_i = (\hat{v}_i, a_i)$, $\boldsymbol{W}_i = (\hat{w}_i, b_i)$,其中 $\hat{v}_i$ 和 $\hat{w}_i$ 称为数据元,a_i 和 b_i 称为增量元。若中间节点收到来自不同源节点的向量并做线性组合时,首先定义新的向量:$\boldsymbol{V}_i' = (\hat{v}_i, a_i, 0)$, $\boldsymbol{W}_i' = (\hat{w}_i, 0, b_i)$,其中 0 代表长为 m 的零向量,所以 $\boldsymbol{V}'_i, \boldsymbol{W}'_i$ 线性组合可计算为 $\boldsymbol{y} = a\boldsymbol{V}' + b\boldsymbol{W}' \in F_q^{n+2m}$,组合后的向量数据元线性组合,而增量元各自保持不变。

2　双源线性网络编码算法

在我们的签名技术中,源节点用自己的私钥为每个向量签名,中间节点首先用公钥验证收到的签名包,然后再做线性组合并用收到的签名计算组合的签名发给下游节点。

2.1　双源网络编码签名算法

双源网络编码签名算法(Network coding Signature, NS)由四个部分组成:参数的生成算法 Setup、签名算法 Sign、组合算法 Combine 和验证算法 Verify。具体实现如下:

Setup 算法

给定安全参数 1^k。

1) 生成一个五元组 (G_1, G_2, H, e, u),其中 G_1, G_2 是两个有着相同素数阶 q 的乘法循环群。

2) 随机选取 $g_1, g_2, \cdots, g_n \leftarrow G_1 \backslash \{1\}$。

3) 随机选择 $u \in G_2 \backslash \{1\}$。

4) 每个源节点建立自己的公私钥对。源节点 $i (i = 1,2)$ 随机选择私钥 $sk_i = \alpha_i \in F_q$,公钥 $pk_i = u_i^{\alpha^{-1}}$。

5) 输出公钥。

Sign 算法

给定私钥 $SK := (\alpha_1, \alpha_2)$。源节点 s_1 计算向量 $\boldsymbol{v}_i = (v_{i1}, v_{i2}, \cdots, v_{im+n})$ 的哈希值:$h_i = \prod_{j=1}^{n} g_j^{v_{ij}}$,用私钥 α_1 计算向量 $\boldsymbol{v}_i$ 的签名 $\sigma_i^1 := h_i^{\alpha_1}$;源节点 s_2 计算向量 $\boldsymbol{w}_i = (w_{i1}, w_{i2}, \cdots, w_{im+n})$ 的哈希值:$h_i = \prod_{j=1}^{n} g_j^{w_{ij}}$,用私钥 α_2 计算向量 $\boldsymbol{w}_i$ 的签名 $\sigma_i^2 := h_i^{\alpha_2}$。

Combine 算法

已知向量:

$\boldsymbol{V}_i' = (\boldsymbol{v}_i, g_i, 0)$, $\boldsymbol{W}_i' = (w_i, 0, b_i)$ 及其对应的签名 σ_i^1, $\sigma_i^2 (i = 1,2,\cdots,m)$,中间节点选取编码向量 $\beta_1, \beta_2, \cdots, \beta_{l_1} \in F_q$, $\gamma_1, \gamma_2, \cdots, \gamma_{l_2} \in F_q$,其中 $l_1 + l_2 = l$,计算向量线性组合:

$$\boldsymbol{y} = \sum_{i=1}^{l_1} \beta_i \boldsymbol{V}_i' + \sum_{i=1}^{l_2} \gamma_i \boldsymbol{W}_i' = (y_1, y_2, \cdots, y_n, y_{n+1}, \cdots, y_{n+m}, y_{n+m+1}, \cdots, y_{n+2m})$$

令 $\beta_i = y_{n+i}, \gamma_i = y_{n+m+i}$,所以:

$\boldsymbol{y} = \sum_{i=1}^{l_1} y_{n+i} \boldsymbol{V}_i' + \sum_{i=1}^{l_2} y_{n+m+i} \boldsymbol{W}_i'$ 组合的签名为:

$$\sigma := \prod_{i=1}^{l_1} (\sigma_i^1)^{y_{n+i}} \prod_{i=1}^{l_2} (\sigma_i^2)^{y_{n+m+i}}$$

Verify 算法

给定公钥:

$PK = (u, g_1, g_2, \cdots, g_n, pk_1, pk_2)$,

$\boldsymbol{y} = (y_1, y_2, \cdots, y_n, y_{n+1}, \cdots, y_{n+m}, y_{n+m+1}, \cdots, y_{n+2m})$

如果:

$$\prod_{i=1}^{l_1} e(\sigma_i^1, pk_1)^{y_{n+i}} \prod_{i=1}^{l_2} e(\sigma_i^2, pk_2)^{y_{n+m+i}} = e\left(\prod_{j=1}^{n} g_j^{y_j}, u\right)$$

输出 1,否则输出 0。

2.2　对 Sign 和 Verify 算法的正确性证明

2.2.1　对源节点产生的签名的验证

对源节点 s_1:

$$e(\sigma_i^1, pk_1) = e\left(\prod_{j=1}^{n} g_j^{v_{ij}}, u\right)$$

因为:

$$e(\sigma_i^1, pk_1) = e\left(\left(\prod_{j=1}^{n} g_j^{v_{ij}}\right)^{\alpha_1}, u^{\alpha_1^{-1}}\right) = e\left(\prod_{j=1}^{n} g_j^{v_{ij}}, u^{\alpha_1 \alpha_1^{-1}}\right) = e\left(\prod_{j=1}^{n} g_j^{v_{ij}}, u\right)$$

对源节点 s_2 的签名验证同上。

2.2.2 对组合后签名的验证

因为：

$$\boldsymbol{y}=\sum_{i=1}^{l_1}y_{n+i}\boldsymbol{V}_i'+\sum_{i=1}^{l_2}y_{n+m+i}\boldsymbol{W}_i'=$$

$$\begin{bmatrix}y_1\\y_2\\\vdots\\y_n\\y_{n+1}\\\vdots\\y_{n+m}\\y_{n+m+1}\\\vdots\\y_{n+m+m}\end{bmatrix}=\begin{bmatrix}\sum_{i=1}^{l_1}\beta_i v_{i1}+\sum_{i=1}^{l_2}\gamma_i w_{i1}\\\sum_{i=1}^{l_1}\beta_i v_{i2}+\sum_{i=1}^{l_2}\gamma_i w_{i2}\\\vdots\\\sum_{i=1}^{l_1}\beta_i v_{in}+\sum_{i=1}^{l_2}\gamma_i w_{in}\\\beta_1\\\vdots\\\beta_m\\\gamma_1\\\vdots\\\gamma_m\end{bmatrix}$$

所以：

$$\prod_{i=1}^{l_1}e(\sigma_i^1,pk_1)^{y_{n+i}}\prod_{i=1}^{l_2}e(\sigma_i^2,pk_2)^{y_{n+m+i}}=$$

$$\prod_{i=1}^{l_1}e\left(\left(\prod_{j=1}^{n}g_j^{v_{ij}}\right)^{\alpha_1},u^{\alpha_1^{-1}}\right)^{y_{n+i}}\times$$

$$\prod_{i=1}^{l_2}e\left(\left(\prod_{j=1}^{n}g_j^{w_{ij}}\right)^{\alpha_2},u^{\alpha_2^{-1}}\right)^{y_{n+m+i}}=$$

$$\prod_{i=1}^{l_1}e\left(\left(\prod_{j=1}^{n}g_j^{v_{ij}}\right),u\right)^{y_{n+i}}\times$$

$$\prod_{i=1}^{l_2}e\left(\left(\prod_{j=1}^{n}g_j^{w_{ij}}\right),u\right)^{y_{n+m+i}}=$$

$$\prod_{j=1}^{n}e\left(g_j^{\sum_{i=1}^{l_1}y_{n+i}v_{ij}},u\right)\prod_{j=1}^{n}e\left(g_j^{\sum_{i=1}^{l_2}y_{n+m+i}w_{ij}},u\right)=$$

$$\prod_{j=1}^{n}e\left(g_j^{\sum_{i=1}^{l_1}y_{n+i}v_{ij}+\sum_{i=1}^{l_2}y_{n+m+i}w_{ij}},u\right)=e\left(\prod_{j=1}^{n}g_j^{y_j},u\right)$$

3 安全性分析

3.1 源节点的攻击

由文献[10]，本文假设在 G 中离散对数问题是不可解的，所以向量哈希函数是安全的，且从源节点的公钥计算其私钥将是不可行的，并且穷举攻击成功的可能性可以忽略。所以任何源节点都不能伪造其他节点的签名。下面证明签名算法(NS)是安全的，即证明本方案也能抵抗中间节点对签名的伪造。

3.2 中间节点的攻击

假设攻击者已经控制了其中一个源节点，由于本算法模型中的哈希函数存在确定性、有效性且输出服从均匀分布，因此我们可以利用随机预言模型来证明该安全性。

下面将证明，只要 Co-CDH 问题无法被破解，本算法就能抵抗来自任意中间节点的签名伪造攻击。

命题 如果存在攻击者 A 成功的伪造签名，那么必存在算法 F(挑战者)可以破解 Co-CDH 问题。

证明 设 g_2 是 G_2 的生成元，挑战者 A 已知 $g_2,u_2\in G_2$，$h\in G_1,u_2=g_2^a$，挑战者 F 的目的是输出 $h^a\in G_1$，再不妨令 $u_1=g_2^{\alpha^{-1}}$，并假设所有逆计算是可行的。

系统设置 挑战者 A 发送生成元 $g_2,r\xleftarrow{R}Z_p$，公钥 $v_1=u_1^{r^{-1}}\in G_2$ 给攻击者 A；

哈希询问 攻击者可在任意时刻向算法询问函数 H 的值，同时，按如下方式建立一个表 *H-list*：$\langle V^{(i)},\omega^{(i)},b^{(i)},c^{(i)}\rangle$，当询问 H 的值时，F 回答如下：

1) 如果询问向量 $\boldsymbol{V}$ 已经在表 *H-list*，则返回 $H(\boldsymbol{V})=\omega\in G_1$；

2) 否则，随机地产生 $c\in\{0,1\}$；

3) 算法 F 选取 $b\xleftarrow{R}Z_p$，如果 $c=0$，计算 $\omega\leftarrow h\cdot\psi(g_2)^b\in G_1$；如果 $c=1$，Φ 计算 $\omega\leftarrow\psi(g_2)^b\in G_1$；

4) 算法 F 把 $\langle V,\omega,b,c\rangle$ 加入到 *H-list* 中，返回 $H(\boldsymbol{V})=\omega$ 给攻击者 A。

签名询问 攻击者 A 在已知挑战者公钥 v_1 的前提下，要求 F 对向量空间 V 中消息 $V_1,V_2,\cdots,V_l$ 的签名进行询问，则算法 F 回答如下：

1) 如果 $c_i=0$，挑战者 A 放弃；

2) 如果 $c_i=1$，则令 $\sigma=\psi(u_2)^{b_i r}$，计算可得：$\sigma=\psi(u_2)^{b_i r}=\omega^{ar}=\omega^{ar}$，所以 σ 是在公钥 $v_1=(u_1)^{r^{-1}}=g_2^{(ar)^{-1}}$ 下的签名。挑战者 F 返回 σ 给攻击者 A。

输出 攻击者最终输出四元组 (pk^*,V^*,σ^*,W^*)。其中 $pk^*=(pk_1,pk_2,\cdots,pk_f)$。其中包含挑战者的公钥 v_1(即存在 $i\in\{1,2,\cdots,f\}$，使得 $v_1=pk_i$，$\sigma^*=(\sigma_1,\sigma_2,\cdots,\sigma_f)$，$W^*=\mathrm{span}(w_1,w_2,\cdots,w_t)$。

对于 $\sigma^*=(\sigma_1,\sigma_2,\cdots,\sigma_f)$，假设前 k 个 $\sigma_1,\sigma_2,\cdots,\sigma_k$ 是对消息 $V_1,V_2,\cdots,V_k(k\leqslant l)$ 的签名，令 σ_w 为 $\sigma^*=(\sigma_1,\sigma_2,\cdots,\sigma_k,\cdots,\sigma_l,\cdots,\sigma_f)$ 中最后 $f-k$ 个元素，是对向量空间 $W^*=span(w_1,w_2,\cdots,w_t)$ 的签名，$\sigma_w=\sigma_{k+1}\cdot\sigma_{k+2}\cdot\cdots\cdot\sigma_{f-k}$ 可看成向量空间 W^* 的聚合签名，满足 $e(\sigma_i,pk_i)=e(\omega_i,g_2)$，对所有的 $i(i=k+1,k+2,\cdots,f-k)$，假设攻击者 A 对消息 V_{k+1} 没有提出询问。

仅当 $c_{k+1}=0,c_i=1(k+2\leqslant i\leqslant f-k)$，挑战者 F 构建一个值：$\sigma_{k+1}\leftarrow\sigma_w\left(\prod_{i=k+2}^{f-k}\sigma_i\right)^{-1}$ 在公钥 v_1 下对某一消息的签名，所以有 $e(\sigma_{k+1},v_1)=e(\omega_{k+1},g_2)$。因为 $c_{k+1}=0$，故 $\omega_{k+1}=h\cdot\psi(g_2)^{b_{k+1}}$，因此 σ_{k+1} 是公钥 $v_1=(u_2)^{r^{-1}}=g_2^{(ar)^{-1}}$ 下对某一哈希值为 $w_{k+1}=h\psi(g_2)^{b_{k+1}}$ 的消息签名，即 $e(\sigma_{k+1},g_2^{(ar)^{-1}})=e(h\cdot\psi(g_2)^{b_{k+1}},g_2)$，所以要求 F 计算并输出 $h^a=\left(\frac{\sigma_{k+1}}{\psi(u_2)^{b_{k+1}}}\right)^{r^{-1}}$。由计算 Co-Deffie-Hellman 问题的困难性可知，攻击者在已知某一公钥的情况下，不能伪造对某一个消息的签名。

3.3 效率分析

本文的签名算法和验证算法与文献[6,8]一样，都需要计算向量的哈希值。在多源网络编码签名算法中，减少中间节点验证过程的运算量是算法的关键点，本文方案由于运用了向量的合并算法，所以减少了中间节点验证过程向量哈希函数的计算量，具体比较如表 1。T_{me} 为计算一个模幂运算所需时间。

(下转第 1520 页)

5 结语

针对分布在网络中的 IDS 通常由不同厂商或组织开发的现状,不同 IDS 没有用于交换知识的共同词汇集,难以交互和协作,对多层次、分布式攻击缺乏检测能力等缺点,本文从语义化的角度,提出了一种基于本体和 OWL 的 Web 服务攻击分类和描述方法,将 Web 服务攻击描述为各 IDS 可以共同理解的包含语义的 Web 服务攻击本体库;在此基础上,设计了一种基于 Web 服务攻击本体库的入侵检测系统(O-IDS),增强了不同 IDS 之间的协作,提高了对多层次、分布式攻击的检测能力。

在后续的研究中,将表征具体 Web 服务攻击的特征抽取出来,作为 Web 服务攻击本体底层子类的属性,完善 Web 服务攻击本体的构建。以完善后的 Web 服务攻击本体库取代本地 IDS 的攻击特征库,完善和实现完全基于 Web 服务攻击本体库的 O-IDS。

参考文献:

[1] PADMANABHUNI S, SINGH V, KUMAR K M S, *et al*. Preventing Service Oriented Denial of Service (PreSODoS) [C]. ICWS'06: Proceedings of the IEEE International Conference on Web Services. Washington, DC: IEEE Computer Society, 2006: 577 - 584.

[2] XU JUN, LEE WOOYONG. Sustaining availability of Web services under distributed denial of service attacks[J]. IEEE Transactions on Computers, 2003, 52(2): 195 - 208.

[3] WANG JUN. Defending against denial of Web services using sessions[R]. Sankt Augustin: NEC Europe, 2006.

[4] 黄康宇, 吴礼发, 吴海佳. Web 服务恶意内容攻击检测技术[J]. 计算机应用, 2010, 30(8): 34 - 38.

[5] 林岳, 宋保华, 段海波, 等. 现代语义技术及其应用[J]. 计算机应用, 2005, 22(6): 130 - 132.

[6] RASKIN V, NIRENBURG S, TRIEZENBERG K E, *et al*. Ontology in information security: A useful theoretical foundation and methodological tool[C]// NSPW'01: Proceedings of the 2001 Workshop on New Security Paradigms. New York: ACM Press, 2002: 67 - 73.

[7] UNDERCOFFER J, JOSHI A, PINKSTON J. Modeling computer attacks: An ontology for intrusion detection[C]// Proceedings of 6th International Symposium on Recent Advances in Intrusion Detection, LNCS 2516. Berlin: Springer-Verlag, 2003: 113 - 135.

[8] OWL Web Ontology Language Guide Recommendation[EB/OL]. [2010 - 09 - 01]. http://www.w3.org/TR/2004/REC-owl-guide-20040210/.

[9] GRUBER T R. A translation approach to portable ontology specifications[J]. Knowledge Acquisition, 1993, 5(2): 199 - 220.

[10] BORST W N. Construction of engineering ontologies for knowledge sharing and reuse[D]. Enschede: University of Tweente, 1997.

[11] STUDER R, BENJAMINS V R, FENSEL D. Knowledge engineering: principles and metholds[J]. Data and Knowlege Engineering, 1998, 25(122): 161 - 197.

[12] NIST. Guide to Secure Web Services[EB/OL]. [2010 - 09 - 01]. http://csrc.nist.gov/publications/nistpubs/800-95/SP800-95.pdf.

[13] 黄康宇, 贺正求, 赖海光, 等. Web 服务攻击技术研究综述[J]. 计算机应用研究, 2010, 27(1): 17 - 22.

[14] OASIS. Web Services Business Process Execution Language Version 2.0[EB/OL]. (2007 - 01 - 31) [2009 - 03 - 24]. http://docs.oasis-open.org/wsbpel/2.0/CS01/wsbpel-v2.0-CS01.pdf.

[15] The common object request broker: Architecture and specification revision 2.0[M]. [S.l.]: QED Publish Company, 1999.

(上接第 1514 页)

表 1 哈希函数运算量对比

方案	签名	验证
文献[6]方案	不计算	$(m+n)T_{me}$
文献[8]方案	$(m+n)T_{me}$	$(m+n)T_{me}$
本文方案	nT_{me}	nT_{me}

4 结语

本文基于安全的哈希函数提出了一个双源网络编码签名算法,该算法证明能够抵御信源节点和中间节点对签名的伪造,但是没有证明对消息伪造的安全性,这也是下一步有待研究的问题。

参考文献:

[1] AHLSWEDE R, CAI N, LI S Y R, *et al*. Network information flow [J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204 - 1216.

[2] KROHN M N, FREEDMAN M J, MAZIERES D. On-the-fly verification of rateless erasure codes for efficient content distribution [C]// IEEE Symposium on Security and Privacy. Oakland: IEEE Computer Society, 2004: 226 - 240.

[3] ZHAO F, KALKER T, MEDARD M, *et al*. Signatures for content distribution with network coding [C]// Proceedings of 2007 IEEE International Symposium on Information Theory. Nice: IEEE, 2007: 556 - 560.

[4] BONEH D, FREEMAN D, KATZ J, *et al*. Signing a linear subspace: Signature schemes for network coding [C]// Public Key Cryptography: PKC-2009, LNCS 5443. Berlin: Springer-Heidelberg, 2009: 68 - 87.

[5] JOHNSON R, MOLNAR D, SONG D, *et al*. Homomorphic signature schemes[C]// Topics in Cryptology— CT-RSA 2002, LNCS 2271. Berlin: Springer-Heidelberg, 2002: 244 - 262.

[6] 罗蛟, 杨铭熙. 安全多源网络编码环签名方案[J/OL]. (2009 - 03 - 03) [2010 - 11 - 23]. http://www.Paper.Edu.cn.

[7] YAN WENJIE, YANG MINGXI, LI LAYUAN, *et al*.. Short signature for multi-source network coding[C]// 2009 International Conference on Multimedia Information Networking and Security. Washington, DC: IEEE Computer Society, 2009: 458 - 462.

[8] CZAP L, VAJDA I. Signatures for multi-source network coding [EB/OL]. (2010 - 06 - 04) [2010 - 11 - 23]. http://eprint.iacr.org/2010/328.

[9] AGRAWAL S, BONEH D, BOYEN X, *et al*. Preventing pollution attacks in multi-source network coding[C]// PKC 2010: Public Key Cryptography. LNCS 6056. Berlin: Springer-Heidelberg, 2010: 161 - 176.

[10] BONEH D, GENTRY C, LYNN B, *et al*. Aggregate and verifiably encrypted signatures from bilinear maps[C]// Advances in Cryptology-Eurocrypt 2003, LNCS 2656. Berlin: Springer-Heidelberg, 2003: 416 - 432.