无线传感器网络的组密钥更新算法 *

张 兴[1,2], 韦 潜[2], 孙旭光[2]
(1. 辽宁工业大学 电子与信息工程学院, 辽宁 锦州 121001; 2. 北京工业大学 计算机学院, 北京 100124)

摘 要: 提出了一种适用于无线传感器网络的基于改进密钥链接树的组密钥更新算法。通过在基于密钥链接树的组密钥管理方案中引入问题密钥路径,并延迟这些问题密钥路径上的密钥更新操作,从而减少密钥链接树中辅助节点上的重复密钥更新。实验结果表明,与现有的组密钥管理方案相比,基于改进密钥链接树的组密钥更新算法在节点添加和删除操作时产生更少的密钥更新消息和消耗更少的能量。

关键词: 无线传感器网络; 组通信安全; 组密钥更新; 密钥链接树
中图分类号: TP212.9 **文献标志码**: A **文章编号**: 1001-3695(2010)01-0223-03
doi:10.3969/j.issn.1001-3695.2010.01.066

Group rekeying algorithm in wireless sensor networks

ZHANG Xing[1,2], WEI Qian[2], SUN Xu-guang[2]
(1. *School of Electronics & Informatics Engineering, Liaoning University of Technology, Jinzhou Liaoning* 121001, *China*; 2. *College of Computer Science & Technology, Beijing University of Technology, Beijing* 100124, *China*)

Abstract: This paper proposed a refined key link-tree based group rekeying algorithm that is suitable for wireless sensor networks. By incorporating problem key paths into the key link tree-based group key management scheme and delaying the key update operations in problem key paths, the number of duplicate key update messages for auxiliary nodes could be reduced. The experimental results show the algorithm requires fewer rekeying messages and costs less power in the node adding and deleting operations than the existing group key management schemes.

Key words: wireless sensor networks; group communication security; group rekey; key link-tree

由于无线传感器网络的资源非常有限,各节点要根据分组算法将大规模网络分成组,以方便数据聚合,减少节点的通信负载,达到延长网络寿命的目的。组成员间通过共享组密钥加/解密采集的数据和组管理信息,一旦有组成员节点被捕获,则所有的相关信息都会泄露。组密钥的及时更新和安全管理在确保大型无线传感器网络的安全通信方面起着至关重要的作用。

1 相关研究现状

文献[1]比较了多组密钥管理协议,指出最好的组密钥管理方案就是使用分级树架构,因为它能在保障安全的同时获取最佳性能。密钥图方法[2,3]就是使用这种架构的典型代表。

1.1 基于逻辑密钥架构的组密钥更新方法

在基于逻辑密钥分层结构的组密钥管理方案中[4],密钥树是一个有向无环图。它包括两种类型的节点,即代表使用者的 u 节点和代表密钥的 k 节点,所有的 u 节点和 k 节点构成一棵树。如果从 u 节点 u_i 到 k 节点 k_j 存在一条有向路径,则将密钥 k_j 分配给使用者 u_i。密钥树的根节点是组密钥,叶子节点是个体密钥,所有的其他节点都是辅助密钥。如果该密钥树的度为 d,树中叶子节点的个数为 N,并且该密钥树是完全平衡的,则添加节点时所产生的密钥更新消息数为 $2\log_d(N)$,删除节点时所产生的密钥更新消息数为 $d\log_d(N)-1$。然而为平衡该密钥树需要产生大量的密钥更新消息。

1.2 基于密钥链接树的组密钥更新方法

密钥链接树是密钥树的一种改进形式,它可以应用于非平衡的树结构中[5]。与密钥树相比,由于密钥链接树无须维护树的平衡状态,具有更好的性能。一棵 d 度密钥链接树 $LT_{d,N}$ 是一个森林,它由一个或多个链接到密钥链接树根节点的子森林 $F_{d,k}$ 构成。每个子森林 $F_{d,k}$ 都是由一棵或多棵度为 d、叶子节点数为 d^k,并且链接到子森林根节点的完整树构成。假设 $N=x_kd^k+x_{k-1}d^{k-1}+\cdots+x_1d+x_0(x_k>0,0\leqslant x_i<d)$,则密钥链接树 $LT_{d,N}$ 中最多包含 $k+1$ 个子森林 $F_{d,k},F_{d,k-1},\cdots,F_{d,0}$。其中,$F_{d,k}$ 包含 $x_kT_{d,k}$ 个节点,$F_{d,k-1}$ 包含 $x_{k-1}T_{d,k-1}$ 个节点,$\cdots$,$F_{d,0}$ 包含 $x_0T_{d,0}$ 个节点。所有子森林都链接到密钥链接树的根节点,它代表所有使用者共享的组密钥,树中的所有叶子节点代表个体密钥,剩余的其他节点代表辅助密钥。

向密钥链接树中添加或删除节点时,需要执行相同大小子森林的合并操作,因此增加了更新组密钥时的总开销。文献[5]中指出,基于密钥链接树的组密钥管理方案在添加节点操作和整体性能上优于基于逻辑密钥分层结构的组密钥管理方案,但基于逻辑密钥分层结构的组密钥管理方案在删除节点操作时则优于基于密钥链接树的组密钥管理方案。为了克服基于密钥链接树的组密钥管理方案在删除节点操作时产生大量密钥更新消息的缺陷,本文通过在密钥链接树中引入问题密钥路径,并延迟这些问题密钥路径上的密钥更新操作,从而减少

收稿日期: 2009-05-24; **修回日期**: 2009-07-12 **基金项目**: 北京市自然科学基金资助项目(KZ200610005003)

作者简介: 张兴(1975-),男,辽宁葫芦岛人,讲师,博士研究生,主要研究方向为无线传感网络安全、计算机网络体系结构(zhang_xing@emails.bjut.edu.cn);韦潜(1970-),男,讲师,博士研究生,主要研究方向为移动 Ad hoc、无线 Mesh 网络安全;孙旭光(1978-),女,讲师,博士研究生,主要研究方向为无线传感网络安全.

了更新组密钥时所需的开销。

2　基于改进密钥链接树的组密钥更新算法

基于改进密钥链接树的组密钥管理方案与基于密钥链接树的组密钥管理方案相比,在组密钥更新过程中产生更少的密钥更新消息。其核心思想是在组成员改变过程中延迟密钥链接树上相应辅助节点的修改,从而减少同一辅助节点上的重复密钥更新操作,进而减少总的密钥更新消息数以及相应的能量开销。文献[6]中指出,攻击者发现无线传感器节点的最基本方法是通过检测传感器节点所释放出的各种信号。部署在邻近区域的传感器节点具有相同的被捕获以及入侵的概率,并且它们被攻击者捕获的顺序以及被网络入侵检测系统识别并从网络中隔离出去的顺序相邻。因此,这些传感器节点应被置于改进密钥链接树的同一子树中,以进一步地减少密钥维护时所需的整体能量消耗。

2.1　节点删除算法

如果改进密钥链接树中一条起始于叶子节点、终止于根节点的路径上的某个辅助节点所对应的密钥信息在叶子节点删除后并没有进行更新,那么称这条路径为问题路径。如果两条问题路径仅相交于根节点,则它们是非关联的;否则,这两条问题路径是相关联的。相关联的问题路径实例如图 1 所示。

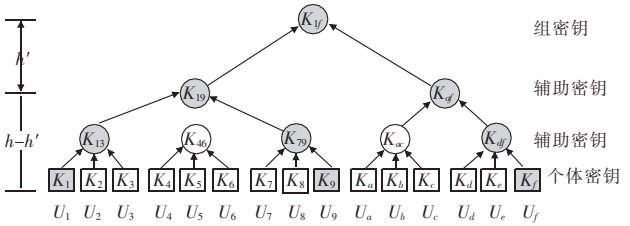

图 1　相关联的问题路径实例

在图 1 的改进密钥链接树中,通过删除叶子节点 U_1,构造一条问题路径 $K_1 \rightarrow K_{13} \rightarrow K_{19} \rightarrow K_{1f}$。新的组密钥可通过以下密钥更新消息进行更新:$E_{K_2}(K')$,$E_{K_3}(K')$,$E_{K_{46}}(K')$,$E_{K_{79}}(K')$,$E_{K_{af}}(K')$。与基于密钥链接树的组密钥管理方案相比,本方案减少了两条密钥更新消息。在改进密钥链接树中构造第一条问题路径时,需要使用问题路径上所有节点的无问题子节点对应的密钥信息更新组密钥。该问题路径上所有节点的子节点数为 $\sum_{i=1}^{h-1} d_i, i \in DP$,其中问题节点个数为 $(h-1)$,因此问题路径所有节点的无问题子节点数为 $\sum_{i=1}^{h-1} d_i - (h-1), i \in DP$。

在图 1 所示的改进密钥链接树中,通过删除叶子节点 U_f 构造第二条问题路径 $K_f \rightarrow K_{df} \rightarrow K_{af} \rightarrow K_{1f}$,它与第一条问题路径相交于根节点 K_{1f},则这两条问题路径是非关联的。新的组密钥 K'' 可以通过以下密钥更新消息进行更新::$E_{K_2}(K'')$,$E_{K_3}(K'')$,$E_{K_{46}}(K'')$,$E_{K_{79}}(K'')$,$E_{K_{ac}}(K'')$,$E_{K_d}(K'')$,$E_{K_e}(K'')$。第一次组密钥更新消息中的 $E_{K_{of}}(K')$ 被 $E_{K_{ac}}(K'')$、$E_{K_d}(K'')$、$E_{K_e}(K'')$ 所替换,因此多了两条密钥更新消息。然而,与基于密钥链接树的组密钥管理方案相比,本方案仍减少了一条密钥更新消息。

在图 1 的改进密钥链接树中,通过删除叶子节点 U_9 构造第三条问题路径 $K_9 \rightarrow K_{79} \rightarrow K_{19} \rightarrow K_{1f}$,它与第一条问题路径相交于非根节点 K_{19},则这两条问题路径是相关联的。新的组密钥 K''' 可通过以下密钥更新消息进行更新:$E_{K_2}(K''')$,$E_{K_3}(K''')$,$E_{K_{46}}(K''')$,$E_{K_7}(K''')$,$E_{K_8}(K''')$,$E_{K_{ac}}(K''')$,$E_{K_d}(K''')$,$E_{K_e}(K''')$。第二次组密钥更新消息中的 $E_{K_{79}}(K'')$ 被 $E_{K_7}(K''')$ 和 $E_{K_8}(K''')$ 所替换,因此多了一条密钥更新消息。与基于密钥链接树的组密钥管理方案相比,本方案仍减少了一条密钥更新消息。

当在改进密钥链接树中构造新的问题路径时,原来的一条密钥更新消息将会被 $\sum_{i=h'+1}^{h-1}(d_i-1), i \in DP' - DP$ 条密钥更新消息所替换,用于更新改进密钥链接树中新的问题路径 $DP' - DP$ 所对应的组密钥信息。基于改进密钥链接树的组密钥管理方案中构造问题路径所需的组密钥更新消息数为

$$f_n = \begin{cases} \sum_{i=1}^{h-1} d_i - (h-1) & n=1 \quad i \in DP \\ f_{n-1} + \sum_{i=h'+1}^{h-1}(d_i-1) - 1 & n>1 \quad i \in DP' - DP \end{cases}$$

其中:n 是改进密钥链接树中问题路径的个数;DP 和 DP' 分别是改进密钥链接树中已有的和新增的问题路径。

基于改进密钥链接树的组密钥管理方案中问题路径越多,所需的组密钥更新消息数就越多;并且问题路径间共享的部分越少,即 h' 越小,所需的组密钥更新消息数就越多。因此,当汇聚于非根节点的问题路径数等于该节点的度时,应刷新问题路径。节点删除算法在刷新汇聚于节点 n 的问题路径时所需的组密钥更新消息数为

$$f_1(n) = \begin{cases} 0 & n = \text{leaf} \\ g_1(n) + f_1(n'\text{s problem children}) & n'\text{s child} = \text{leaf} \\ g_2(n) + f_1(n'\text{s problem children}) & n = \text{others} \end{cases}$$

其中:$g_1(n)$ 和 $g_2(n)$ 分别为节点 n 的无问题子节点个数和子节点个数。节点删除算法在更新以节点 n 为根的子树时所需的组密钥更新消息数为

$$f_2(n) = \begin{cases} g_1(n) + f_2(n'\text{s problem children}) & n = \text{others} \\ 0 & n = \text{leaf} \end{cases}$$

其中:$g_1(n)$ 是节点 n 的无问题子节点个数。节点删除算法所需的组密钥更新消息数为 $f_{\text{evicting}} = f_1(n) + f_2(\text{root})$。其中:改进密钥链接树中汇聚于节点 n 的非关联问题路径数等于节点 n 的度,树中最多只存在一个这样的节点;root 是改进密钥链接树的根节点。

2.2　节点添加算法

当向无线传感器网络中添加新的传感器节点时,从改进密钥链接树中选择一条问题路径,并将此节点放置在问题路径的叶子节点上,同时刷新问题路径直至遇到有多条问题路径汇聚的第一个节点为止。因此,刷新问题路径所需的组密钥更新消息数等于问题路径上始于叶子节点、终于多条问题路径汇聚的第一个节点的相应部分上所有问题节点的度之和,即 $\sum_{i=h'+1}^{h-1} d_i, i \in DP' - DP$。当更新改进密钥链接树中相同部分的组密钥而不刷新问题路径时,只需更新这些相同节点的非问题子节点以及新加入的节点所分配的组密钥信息。因此,将会节省问题路径上始于叶子节点的父节点、终于多条问题路径汇聚的第一个节点的相应部分上的 $(h-1)-h'$ 条密钥更新消息。

在刷新问题路径之后,改进密钥链接树中对应子树上的所有节点都处于无问题状态,因此更新该子树只需一条组密钥更新消息。反之,不刷新问题路径时更新该对应子树所需要的组密钥更新消息数等于原始的密钥更新消息数 $\sum_{i=h'+1}^{h-1}(d_i-1)$ 加上更新节点时的密钥消息数 1,即 $\sum_{i=h'+1}^{h-1}(d_i-1)+1, i \in DP' - DP$。在改进密钥链接树中刷新问题路径可以节省 $\sum_{i=h'+1}^{h-1}(d_i-1)$ 条密钥更新消息。因为当 $d_i \geqslant 3$ 时,$\sum_{i=h'+1}^{h-1}(d_i-1) > > (h-1)-h'$,所以在刷新问题路径时应选择 h' 最小的问题路径进行刷新。

当改进密钥链接树中不存在问题路径时,应选择链接树中空属性值为真的叶子节点进行插入。如果树中不存在这样的节点,则改进密钥链接树就变成了密钥链接树,此时应使用基

于密钥链接树的组密钥管理方案中的节点添加算法向树中插入新的节点。在基于改进密钥链接树的组密钥管理方案中,节点添加算法所需的组密钥更新消息数为

$$f_{\text{adding}}=\begin{cases}\sum_{i=h'+1}^{h-1}d_i+f_2(\text{root}) & \text{存在问题路径},i\in DP'-DP\\ 2(h-1) & \text{不存在问题路径,存在空节点}\\ C_{\text{Add}}^{\text{LTP}}(N,d) & \text{既不存在问题路径,也不存在空节点}\end{cases}$$

其中:$C_{\text{Add}}^{\text{LTP}}(N,d)$是基于密钥链接树的组密钥管理方案中的节点添加算法所需的组密钥更新消息数[5]。

$$C_{\text{Add}}^{\text{LTP}}(N,d)=2+2\times f_1(N,d)$$

$$f_1(N,d)=\begin{cases}0 & \text{if } x_0=0\\ 1 & \text{if } 0<x_0<d-1\\ i+1 & \text{if } x_i=x_{i-1}=\cdots=x_0=d-1,x_{i+1}=0\\ i+2 & \text{if } x_i=x_{i-1}=\cdots=x_0=d-1,0<x_{i+1}<d-1\\ k & \text{if } x_{k-1}=x_{k-2}=\cdots=x_0=d-1\end{cases}$$

3　性能分析

本文使用构建的模拟器来比较基于改进密钥链接树的组密钥管理方案和其他组密钥管理方案的性能,每次实验运行 1 000次并取平均值,以便得到合理且稳定的实验结果。图 2 展示了在树的度为 3 时,基于改进密钥链接树的组密钥管理方案、基于密钥链接树的组密钥管理方案以及基于逻辑密钥分层结构的组密钥管理方案在删除一个节点时各自所需的密钥更新消息数。从图中可以看出,基于改进密钥链接树的组密钥管理方案产生最少的密钥更新消息,而基于密钥链接树的组密钥管理方案产生最多的密钥更新消息,并且这三种方案在删除一个节点时所需的组密钥更新消息数均随着树中节点数的增加而增加。

图 3 展示了基于改进密钥链接树的组密钥管理方案和基于密钥链接树的组密钥管理方案在叶子节点数 nodes = 50、度 $d=3$、操作数 $n=25$ 的条件下按不同比例随机添加节点时所需的密钥更新消息数。从图中可以看出,基于改进密钥链接树的组密钥管理方案比基于密钥链接树的组密钥管理方案产生最少的密钥更新消息。

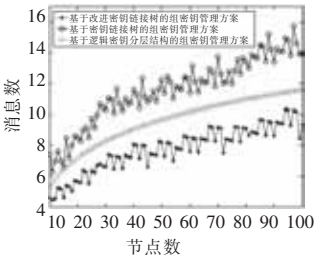

图 2　删除一个节点时所需的更新消息数

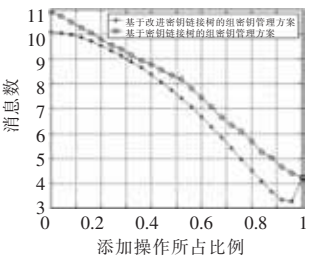

图 3　随机添加节点时所需的更新消息数

4　结束语

本文提出了基于改进密钥链接树的组密钥更新算法,它可以在无线传感器网络的所有节点间维护一个安全的组密钥。实验结果表明,与现有的组密钥管理方案相比,基于改进密钥链接树的组密钥更新算法在添加和删除节点时产生更少的密钥更新消息,因此消耗节点更少的能量,能够维持更长的网络寿命。

参考文献:

[1] RAFAELI S, HUTCHISON D. A survey of key management for secure group communication[J]. ACM Computing Surveys, 2003, 35(3): 309-329.

[2] WONG C K, GOUDA M, LAM S S. Secure group communications using key graphs[C]//Proc of ACM SIGCOMM'98. 1998.

[3] WALDVOGEL M, CARONNI G, SUN Dan, *et al*. The versakey framework: versatile group key management[J]. IEEE Journal on Selected Areas in Communications, 1999, 17(9): 1614-1631.

[4] 徐勇,陈恺. 安全多播中基于成员行为的 LKH 方法[J]. 软件学报, 2005, 16(4): 601-608.

[5] ZHANG Jun, ZHOU Yu, MA Fan-yuan, *et al*. An extension of secure group communication using key graph[J]. Information Sciences, 2006, 176(20): 3060-3078.

[6] GU Wen-jun, WANG Xun, CHELLAPPAN S, *et al*. Defending against search-based physical attacks in sensor networks[C]//Proc of the 2nd IEEE International Conference on Mobile Ad hoc and Sensor Systems. Washington DC: IEEE Press, 2005: 520-527.

(上接第 222 页)

3.3　可恢复性实验

在图像通过精确认证时,用 2.4 节所述的方法擦除水印,恢复出原始图像。实验过程中,通过计算原始图像与擦除水印后恢复的原始载体图像 PSNR 值来判断恢复图像的准确性。实验证明,PSNR 值为无穷大,表明恢复图像与原始图像完全相同,算法的可恢复性好。

4　结束语

本文针对 Tian 算法存在过分修改像素对灰度值、算法分类复杂、须嵌入二值定位图等缺点,提出一种基于双分量差值扩展的彩色图像可擦除水印算法。该算法将差值扩展量分散到四个灰度值中,减少了对图像的修改,并从理论和实验两方面证明了含印图像质量明显提高。嵌入像素对分类简单,容错数据的处理方法比压缩二值定位图简单得多,且不影响水印容量,算法实现和计算时间都具有优势。实验表明,该算法在保证水印透明性的基础上,可以逐对嵌入水印,实现了水印的盲提取,并在水印擦除后可以完全恢复原始图像,适合于彩色图像的精确认证和窜改定位,定位精度达到 2 像素级。该算法在透明性和水印容量方面取得了较好的效果,在军事、法律和医学等领域具有广泛的应用前景。下一步的工作是在另一色彩分量中嵌入检测水印,以提高定位水印的安全性。

参考文献:

[1] COX I J. 数字水印[M]. 王颖,黄志蓓,译. 北京: 电子工业出版社, 2003: 216-241.

[2] 刘向丽,刘向阳,寇卫东. 利用差分扩展实现可逆水印[J]. 光电子· 激光, 2008, 19(8): 1094-1096.

[3] TIAN Jun. Reversible data embedding using a difference expansion [J]. IEEE Trans on Circuits and Systems for Video Technology, 2003, 13(8): 890-896.

[4] ALATTAR A M. Reversible watermark using the difference expansion of a generalized integer transform[J]. IEEE Trans on Image Processing, 2004, 13(8): 1147-1156.

[5] 邓世文,刘焕平,叶宏宇. 基于 Laplacian 残差扩展的可逆嵌入算法[J]. 计算机工程与应用, 2008, 44(3): 110-113.

[6] 彭德云,王嘉祯. 基于错误控制编码的差值扩展可逆数字水印[J]. 计算机工程, 2007, 33(21): 18-20.

[7] 祝玉新,孙星明,杨恒伏. 基于 Haar 小波的彩色图像可逆水印算法[J]. 计算机应用研究, 2007, 24(6): 165-166, 169.

[8] 曹文伦,彭国华,秦洪元,等. 利用色彩分量相关性的彩色图像分形编码方法[J]. 计算机工程与应用, 2004, 40(22): 51-55.

[9] 王丽娜,张焕国. 信息隐藏技术与应用[M]. 武汉: 武汉大学出版社, 2003: 57-70.