ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.7, July 2010, pp.1679−1691
doi: 10.3724/SP.J.1001.2010.03585

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

无线传感器网络动态密钥管理方法*

孔繁瑞+, 李春文

(清华大学 自动化系,北京 100084)

Dynamic Key Management Scheme for Wireless Sensor Network

KONG Fan-Rui+, LI Chun-Wen

(Department of Automation, Tsinghua University, Beijing 100084, China)

+ Corresponding author: E-mail: kongfr@mails.thu.edu.cn

Kong FR, Li CW. Dynamic key management scheme for wireless sensor network. *Journal of Software*, 2010,21(7):1679−1691. http://www.jos.org.cn/1000-9825/3585.htm

Abstract: Key management is a critical issue for wireless sensor networks security. This paper proposes EEHS, a novel Energy Efficient and Highly Survivable dynamic key management scheme for large-scale clustered wireless sensor networks based on Exclusion Basis System (EBS). The major advantages of EEHS are strengthened network security, enhanced energy efficiency, high dynamic performance and good support for network expansion. In EEHS, a novel polynomial-based key—the common polynomial key, is designed and employed as the administration key in the EBS, which can enhance the network survivability under attack. All system keys can be refreshed and revoked according to the compromise of sensor nodes. The function of key assignment and key generation are dispatched to different functional nodes in one cluster and sensor nodes also rotate to act as functional nodes in order to improve the energy efficiency and the network robustness. Simulation and analysis results show that compared with related works, EEHS supports the networks with more energy efficiency, longer lifespan and stronger resilience to node compromise.

Key words: wireless sensor networks; security; EBS (exclusion basis system)-based key management; common polynomial key; energy efficiency

摘 要: 设计安全、合理的密钥管理方法,是解决无线传感器网络安全性问题的核心内容.提出了 EEHS(a novel energy efficient and highly survivable dynamic key management scheme),是一种基于 EBS(exclusion basis system)的、适合于大规模分簇式无线传感器网络的动态密钥管理方法,它具有安全性强、能量效率高、动态性能好、可扩展性强等特点.一种新型多项式密钥(同化多项式密钥)被运用在了 EEHS 的 EBS 管理密钥中,显著地提高了网络的抗捕获能力.当节点被捕获时,EEHS 还可以动态取消并更新被捕获节点所拥有的全部密钥,最终驱逐被捕获的节点.为提高网络的能量效率和鲁棒性,EEHS 将密钥分配和密钥生成等功能分配给簇内不同的功能节点,且传感器节点轮流作为功能节点使用.仿真与分析结果表明,与相关工作相比,EEHS 可以显著提高网络的能量效率、延长网络寿命、加强网络安全性.

关键词: 无线传感器网络;安全性;基于 EBS 的密钥管理;同化多项式密钥;能量有效性

* Supported by the National Natural Science Foundation of China under Grant Nos.69774011, 60433050 (国家自然科学基金)
Received 2008-07-23; Accepted 2009-02-16

中图法分类号: TP393 文献标识码: A

无线传感器网络(wireless sensor networks,简称 WSNs)的应用十分广泛,涵盖了众多领域[1].在很多军事应用中,WSNs 常常工作在无人监管的作战区域,网络内的通信会被监听,传感器节点也非常容易被捕获、篡改和破坏.所以,保证数据的完整性、准确性显得尤为重要.而在一些民事应用中,例如智能大厦、网络数据包含的用户个人信息应受到保护、避免泄露,这需要对数据的私密性进行保护.因此,近年来,WSNs 的网络安全问题得到了广泛的重视.由于 WSNs 的网络拓扑结构具有随机性,传感器节点能量、计算能力、存储空间、通信资源有限,传统的具有 Internet 特色的安全体系——基于第三方的公共密钥安全体系并不适合 WSNs,甚至根本无法使用[2].2002 年,Eschenauer 和 Gligor 首次提出密钥预分配方法[3],而后,在此基础上形成了很多无线传感器网络密钥管理方法.这些方法具有相似的特点,都是在一个固定的密钥池中分配给各个节点一些固定的组合,是一种静态的密钥管理方法.2006 年,Eltoweissy 在 Exclusion Basis Systems(EBS)[4]和传感器网络的分簇结构基础上提出了动态密钥管理的概念[5],它与静态密钥管理相比,其主要优点表现在[5]:

(1) 网络规模不受节点存储空间限制,适合于大规模分簇式网络;

(2) 可以动态而且高效地取消任意节点所拥有的全部密钥,从而驱逐被敌人捕获的节点,提高了网络的安全性能;

(3) 在提供同等安全性保证的条件下,相比于静态密钥管理,既节约了存储空间,又提高了能量效率.

因此,基于 EBS 的动态密钥管理方法也成为无线传感器网络动态密钥管理研究的基础[6−8].本文提出的 EEHS(energy efficient and highly survivable)是一种基于 EBS 的、适合于大规模分簇式无线传感器网络的动态密钥管理方法,它具有安全性强、能量效率高、动态性能好、可扩展性强等特点.

1 EBS 和基于 EBS 的无线传感器网络动态密钥管理方法

EBS 是由 Eltoweissy 等人于 2004 年提出的一种基于组合原理的组通信密钥管理方法[4].基于 EBS 的无线传感器网络动态密钥管理方法中共有两种密钥:管理密钥和会话密钥.管理密钥又被称为密钥生成密钥,它组成了 EBS 密钥体系,但管理密钥并不直接用于通信数据的加密,而主要用于 EBS 内部的密钥管理事件,包括密钥系统的建立和更新、生成会话密钥、驱逐节点等;会话密钥又被称为通信密钥,当 EBS 系统建立以后,会在线地生成会话密钥,用于组内或某些特殊节点之间的通信数据加密.

定义 1(*EBS*(*n*,*k*,*m*)). 设 n,k,m 均为正整数,且 $1<k,m<n$.*EBS*(n,k,m)是以集合$\{1,2,\ldots,n\}$的子集为元素构成的集合Γ,并且对于$\forall t\in\{1,2,\ldots,n\}$,满足以下两个条件:

(1) t 最多出现在Γ的 k 个元素中;

(2) Γ中恰好有 m 个元素 $A_1,A_2,\ldots,A_m$,它们的并集 $\bigcup_{i=1}^{m}A_i=\{1,2,...,n\}-\{t\}$ (意味着任何一个用户 t 都可以由恰好 m 个集合排斥掉).

在基于 *EBS*(n,k,m)的无线传感器网络动态密钥管理方法中,n 表示节点数目,k 表示分配给每个节点的管理密钥个数,$k+m$ 表示管理密钥总数.可以证明[4]:

(1) 当 $\binom{k+m}{k}\geqslant n$ 时,$\binom{k+m}{k}$ 中的任意 n 个组合方式均可以构成 *EBS*(n,k,m),进而形成一个管理密钥的分配方案;

(2) 通过广播最多 m 个数据包,可以取消并更新任意节点拥有的全部管理密钥,从而驱逐该节点.

关于 EBS 结构的详细知识可参考文献[4].

2 同化多项式密钥

定义 2(同化多项式). $f(x_1,x_2,x_3)$为 $t+1$ 阶同化多项式,若 $f(x_1,x_2,x_3)=C+\sum_{i_1=1}^{t+1}\sum_{i_2=1}^{t+1}\sum_{i_3=1}^{t+1}a_{i_1i_2i_3}x_1^{i_1}x_2^{i_2}(x_3-x_c)^{i_3}$,其中,$x_c$

为一常数,$C,a_{i_1i_2i_3}$ 属于有限域 F_q,q 为一个可以容纳管理密钥的足够大的质数.

显然,同化多项式具有一条重要的性质:$\forall x_1,x_2,f(x_1,x_2,x_c)=C$.意味着只要变量 x_3 取值为 x_c 时,多项式 $f(x_1,x_2,x_3)$ 将得到一个常数.在布置网络之前,可以将节点的 ID 设定为一个二元组,因此,任意被分配了同化多项式 f 的节点 N_i 把自己的 ID 作为变量 x_1,x_2 带入 f 中,都将得到一个形如 $f(ID_{i_1},ID_{i_2},x_c)$ 的 t+1 阶一元多项式并保存起来.当一组被分配了同一个同化多项式的节点把 x_c 带入存储在其上的一元多项式时,便可以得到一个共享密钥 $key=f(ID_1,ID_2,x_c)=C$.

与普通的多项式密钥相比,最主要的区别在于,普通多项式密钥只能形成两个节点之间的对密钥,而同化多项式密钥可以形成多节点之间的共享密钥.除此之外,同化多项式密钥具有更好的抗捕获性能.可以证明,t+1 阶同化多项式密钥是 t^2-安全的.即,当捕获不超过 t^2 个拥有某一同化多项式的节点时,该多项式仍是不可破解的,利用该多项式得到的共享密钥仍是安全的.除此之外,由于在 EBS 的很多密钥管理事件中都需要对一组节点进行广播,而普通的多项式密钥只能由两两节点之间私有,所以在广播过程中会导致广播包数目随着节点数目而线性增加;而同化多项式密钥的优点在于形成多节点之间的共享密钥,可以使广播包数目不随节点数目而发生改变,从而节约了传感器节点宝贵的能量资源.

3 系统模型与假设

3.1 网络模型与假设

首先在网络拓扑结构方面,相比于平面结构,分簇式结构更适合大规模的组网,能量有效性也更高.除此之外,分簇式的结构还能够将网络受到的攻击限制在局部区域之内,提高了网络安全性能.在 EEHS 中采用的是分簇式网络结构.其中的节点按照功能可以划分为 3 类:

(1) 传感节点 SN(sensing node),这些节点负责完成网络的基本任务,例如环境监测、人员定位等.它们将获得的外界数据进行初步处理,然后发送至自己所在簇的簇头节点;

(2) 簇头节点 HN(head node),这些节点是簇内的管理者.在数据收集方面,簇头节点是传感节点收集数据的汇聚点,并对数据进行深度分析、融合压缩,再将结果发送至远端的数据终端(data terminal,简称 DT).而在网络安全方面,簇头节点负责密钥分配、更新密钥、接收节点、驱逐节点、共谋后恢复等功能;

(3) 密钥生成节点 KGN(key generation node),这些节点负责生成管理密钥和共谋恢复密钥.

3.2 外界攻击模型与假设

工作在无人监管环境下的 WSNs 面临的攻击种类繁多,例如监听攻击、洪泛攻击、虫洞攻击等.其中,对于密钥体系威胁最大的是节点捕获攻击,因为通过捕获节点,攻击者可以获得节点存储空间中的全部数据,包括密钥.而随着被捕获节点的增加,它们共享得到的密钥信息直至所有管理密钥均被捕获时,整个密钥系统完全丧失了安全性.这种情况被称为共谋,也是基于 EBS 的密钥系统的一个核心问题[4,9].

在外界攻击方面,我们主要作了如下假设:

(1) 网络受到的攻击主要为节点捕获攻击,且被捕获的节点之间可以共享信息;节点被捕获后,其上的全部密钥也同时被捕获;

(2) 节点捕获分为可识别的捕获和不可识别的捕获两种.系统中设计了一些攻击检测功能 IDS(intrusion detection system),利用这些功能可以识别部分节点捕获攻击,进而进行密钥恢复;

(3) 任何节点都有被捕获的可能,包括 SN,HN 和 KGN;数据终端 DT 是安全的;

(4) 网络初始化阶段是安全的,即节点不会在这个期间被捕获.

4 EEHS 的完整描述

4.1 节点初始化

在布置网络之前,需要对节点进行初始化.在 EEHS 中,要求节点有一个全网唯一的 ID、一个与 DT 之间的私有密钥 K 用于确认节点的合法身份,一个统一的单向密钥生成函数 F.表征节点状态的变量也同时需要初始化,包括分簇状态 *Cluster_State* 初始化为 *SN_Unclustered*;自选为 HN 的概率 p_{self} 初始化为 p_{ini};所在簇的 HN 的 ID_c 初始化为(0,0);距离所在簇的 HN 的跳数 H_c 初始化为 *HTS*.

4.2 网络分簇结构初始化

传感节点 SN 被随机地布置在监测区域后,需要自主地进行分簇组网.为了保证簇头节点 HN 尽量分布均匀以平衡网络内的能量消耗,我们采用多轮次的分簇方式,而非一次性确定网络结构.每一个轮次耗时 t_{cr},以保证 HN 的分簇邀请包能够传递至簇的最大半径 *HTS* 跳范围.

EEHS 的分簇过程中每个轮次可以分为以下几个主要步骤:

(1) 状态为 *SN_Unclustered* 的节点生成一个(0,1)之间的随机数 *rand*,如 $rand<p_{self}$ 则自选为 HN,设定自身状态为 *HN_Elected*;

(2) HN 广播分簇邀请包,以多跳的方式邀请处于其 *HTS* 跳范围内的 SN 加入它的簇.分簇邀请包包括 HN 的 ID、被转发的次数 H(初始值为 0)两个部分;

(3) 收到分簇邀请包的 SN 节点记录分簇邀请包中的 ID 和 H.若 $H\geqslant HTS$,则丢弃该包,否则,设定自身状态为 *SN_Clustered*;若 $H<H_c$,则令 $ID_c=ID$,$H_c=H$,修改分簇邀请包的 H 参数(H 自加 1)并广播修改后的分簇邀请包;若 $H\geqslant H_c$,则丢弃该包;

(4) 经过 t_{cr} 时间后,状态仍为 *SN_Unclustered* 的节点将 p_{self} 扩大两倍,即新的 $p'_{self}=2p_{self}$,而后进行新一个轮次的循环.

4.3 EBS密钥系统初始化

网络布置完毕并建立了分簇结构后,需要立即建立 EBS 密钥系统.这一过程可分为节点注册、生成管理密钥、分配管理密钥、初始化会话密钥、初始化共谋恢复密钥这 5 个部分.

4.3.1 节点注册

网络分簇结构建立后,首先,簇内的 SN 节点应向它们的 HN 节点进行安全注册,注册信息包括 SN 的 ID 和 H_c.然后,HN 节点向 DT 进行安全注册,注册信息包括 HN 自身的 ID 及其簇内全部 SN 的 ID 和 H_c.这个过程包括以下几个部分:

(1) HN 向 DT 申请用于节点注册的种子,该申请信息由 HN 利用自身的 K 加密,内容包括 HN 的 ID;DT 记录申请信息,并生成种子 S_{rd},由 K 加密发送至 HN;HN 将 S_{rd} 在簇内广播;

(2) SN 收到 S_{rd} 后,将它作为参数带入函数 F 中得到注册密钥 $K_{sr}=F(S_{rd})$,然后利用 K_{sr} 加密 SN 的注册信息发送至 HN;

(3) HN 同样通过 S_{rd} 和 F 得到 K_{sr},然后解密 SN 的注册信息,确认 SN 的合法身份并记录 SN 的 ID 和 H_c;

(4) HN 利用自己与 DT 之间的私有密钥 K 加密 HN 注册信息向 DT 注册.

利用函数 F 生成 K_{sr} 来加密 SN 注册信息的目的是防止伪装节点攻击,因为伪装节点无法获得 F,所以无法得到 K_{sr},故不能进行安全注册.HN 注册信息中包括簇内全部 SN 的 ID 和 H_c,是因为很多 IDS 系统需要这些信息进行网络异常判断,以提高攻击识别的准确度.

4.3.2 生成管理密钥

HN 在节点注册阶段可以获得簇内的节点数目 n,根据 n 选择合适的 EBS 参数 k,m.首先,必须保证 $\binom{k+m}{k}\geqslant n$;除此之外,$k$,$m$ 之间的关系决定了能耗与安全性之间的平衡[4],它们之间的关系应根据网络的功能

和所处的环境安全性而定.HN 设定好 EBS 结构后,将在簇内选择 KGN,由 KGN 生成管理密钥,并将这些密钥发送给 HN.这样做的目的是将密钥分配和密钥生成两个功能分配给不同的节点完成,以防止 HN 被捕获导致整个密钥体系被破解.KGN 的数目根据网络所处区域的安全状况而定,但为了保证当 KGN 被捕获后密钥系统能够顺利恢复,每个 KGN 生成的密钥数不应该超过 k 个,即 KGN 的数目至少为 $\left\lceil \frac{k+m}{k} \right\rceil$ 个.这个过程分为以下几个部分来完成:

(1) HN 向 DT 申请用于 HN 与 KGN 间通信的密钥种子,该申请信息由 HN 利用自身的 K 加密,内容包括 HN 的 ID;DT 记录申请信息并生成种子 S_{gd},由 K 加密发送至 HN;

(2) HN 将 S_{gd} 和需要生成的管理密钥数目发送给 KGN;

(3) KGN 生成管理密钥,并用 $K_{kg}=F(S_{gd})$加密这些管理密钥发送给 HN.

4.3.3 分配管理密钥

获得 KGN 生成的管理密钥后,HN 将把它们分配给簇内的所有 SN.首先,HN 向簇内广播 EBS 矩阵,使得所有 SN 获得自己的密钥分配方案;而后,将 KGN 生成的全部管理密钥逐个地在簇内广播;最后,将这些管理密钥销毁.SN 收到管理密钥后,根据 EBS 矩阵得到自己的 k 个管理密钥而忽略其他的 m 个,最后销毁 EBS 矩阵.这个过程分为以下几个部分来完成:

(1) HN 向 DT 申请用于分配管理密钥的种子,该申请信息由 HN 利用自身的 K 加密,内容包括 HN 的 ID;DT 记录申请信息并生成种子 S_{dd},由 K 加密发送至 HN;HN 向簇内广播用于分配管理密钥的种子 S_{dd},并利用密钥 $K_{kd}=F(S_{dd})$加密 EBS 矩阵在簇内广播;

(2) SN 利用 S_{dd} 和 F 得到 K_{kd},进而解密得到 EBS 矩阵;

(3) HN 逐个将 $k+m$ 个管理密钥向簇内广播,这些广播包同样由 K_{kd} 进行加密;

(4) SN 按照 EBS 矩阵中自己的密钥分配方案得到属于自己的 k 个管理密钥;

(5) HN 销毁全部 $k+m$ 个管理密钥和函数 F;

(6) SN 销毁 EBS 矩阵.

4.3.4 初始化会话密钥

会话密钥用于簇内的数据通信,可以是全簇唯一的,也可以是按照需要存在于某两个或某些节点之间的.在网络的运行过程中,为提高数据安全性,还会定期或按需地更新会话密钥.在网络初始化时,可以先生成一个簇内唯一的会话密钥 K_{conv},HN 利用管理密钥进行加密后在簇内广播,SN 收到数据包后解密得到 K_{conv}.

4.3.5 初始化共谋恢复密钥

当共谋发生时,即攻击者通过捕获节点获得了所有 $k+m$ 个管理密钥时,需要进行系统恢复,在未被捕获的节点之间重新建立密钥体系,但同时又要保证被捕获的节点被隔离在重建过程之外.EEHS 中采用一种根据 SN 和 HN 之间距离进行分类的方法进行共谋后的恢复,该恢复方法中,需要在不同距离的 SN 和 HN 之间建立唯一的密钥,该密钥被称为共谋恢复密钥 K_{cr}.KGN 负责生成所有的共谋恢复密钥 K_{cr},以 K_{conv} 加密后发送给 HN.HN 为 H_c 相同的一组 SN 转发相应的以 K_{conv} 加密的 K_{cr},并将这些 K_{cr} 发送给 DT.

至此,EBS 密钥系统的初始化全部完成,所有 SN 节点得到了自己的 k 个管理密钥、会话密钥和与自己的 H_c 相对应的共谋恢复密钥;KGN 节点负责生成和保存密钥,但并不知道它所生成的密钥分配给了哪些节点;HN 节点只拥有会话密钥,同时保存 EBS 矩阵,即密钥分配方案,但并不保存实际的管理密钥.整个过程中的通信都有相应的密钥(K_{kg},K_{sr} 等)进行加密,从而保证了数据的安全性.

4.4 EEHS在常态下的功能

在常态下,即未受到攻击或者未检测到攻击时,EEHS 主要包括密钥更新、添加新节点和功能节点轮换 3 种功能.

4.4.1 密钥更新

为提高网络的安全性能,基于 EBS 的密钥系统会在网络的运行过程中周期性或是按需地进行密钥更新,包

括管理密钥、会话密钥和共谋恢复密钥.HN 向 KGN 发送更新密钥请求,KGN 收到后生成新的管理密钥 K'_a 和共谋恢复密钥 K'_{cr},并且利用现有的 K_{kg} 和函数 F 计算新的 $K'_{kg}=F(K_{kg})$,并将 $K'_{kg}(K'_a,K'_{cr})$ 发送给 HN.而 HN 由于在初始化阶段已经销毁了函数 F,因此不能通过 K'_{kg} 解密到 K'_a 和 K'_{cr},只能将由 $K'_{kg}(K'_a,K'_{cr})$ 逐一在簇内广播.SN 同样利用现有的 K_{kg} 和函数 F 得到 K'_{kg},解密 HN 转发的 $K'_{kg}(K'_a,K'_{cr})$ 后得到属于自己的 K'_a 和 K'_{cr}.更新会话密钥过程与初始化会话密钥过程相似,只是用旧的会话密钥取代管理密钥对新的会话密钥进行加密.

在网络的运行过程中,HN 可能被攻击者捕获,因此在初始化阶段销毁函数 F 使得 HN 不能通过解密 $K'_{kg}(K'_a,K'_{cr})$ 得到 K'_a 和 K'_{cr},从而防止攻击者通过捕获 HN 得到系统的全部密钥.

4.4.2 添加新节点

在使用寿命较长的无线传感器网络中,节点会因能量枯竭而失效.因此,为保持一定的节点密度,会有节点在网络运行过程中加入.在新节点被布置之前,DT 向各个 HN 发送由 K 加密的将要布置在网络区域内的新节点的 ID,以防止攻击者实施伪造攻击.新节点布置后,HN 在簇内广播邀请新节点加入,邀请内容包含 HN 的 ID 和记录邀请被转发次数的变量 H_D,并将 H_D 初始化为 1.SN 收到邀请后,若 ID_C=ID_H 且 H_c=H_D,则将 H_D 增大 1 后把邀请转发出去.新节点在所有收到的邀请中选择 H_D 最小的簇加入,并向其 HN 发送自己的 ID 和 H_c 进行注册.

新节点注册完毕后,HN 根据簇内的最新节点数目 n 判断是否需要改变现有的 EBS 结构.若 $n\leqslant\binom{k+m}{k}$,则无需改变现有的 EBS 结构,只需在 $\binom{k+m}{k}$ 中寻找一个尚未被 EBS 矩阵使用的组合,再从 KGN 处得到由 K'_{kg} 加密的管理密钥、共谋恢复密钥和生成此 K'_{kg} 的种子 K_{kg},连同会话密钥一起转发给新节点;若 $n>\binom{k+m}{k}$,则将调整现有的 EBS 结构适应节点数目的增加,然后按照第 4.3 节中的方法重新初始化 EBS 密钥体系,以使网络在新的 EBS 下运行.

4.4.3 功能节点轮换

簇内的功能节点包括 HN 和 KGN 两种,它们担负着数据汇聚、密钥管理等功能,相比于 SN 消耗的能量更多.所以在 EEHS 中,功能节点由簇内节点轮流担任,这样可以平衡节点的能量消耗,延长网络寿命.除此之外,设计功能节点轮换机制还可以高效地进行功能节点被捕获后的恢复.

我们采用与 LEACH(low-energy adaptive clustering hierarchy)[10]中类似的功能节点选举机制,但为了保证每一轮次中都有确定数目的功能节点被选出来,还附加了一些功能节点数目的控制环节.选举过程中,节点第 r 轮当选的概率设为 $T(r)$[10]:

$$T(r)=\begin{cases}\dfrac{P}{1-P\times\left(r \bmod \dfrac{1}{p}\right)}, & i\in G\\ 0, & \text{否则}\end{cases},$$

其中,$P=\dfrac{n_F}{n}$ 为功能节点在簇内所占的比例,r 是轮次,G 为过去 r–1 轮中还未当选的节点.这种选举方式的优点在于,可以保证 $\dfrac{1}{p}$ 轮之内每个节点最多当选 1 次,既充分均衡了簇内的能量消耗,又可以防止伪装成功能节点的攻击.

功能节点轮换过程可概括为以下几个步骤:

(1) 旧 HN 向簇内发出功能节点轮换通知;

(2) SN 收到轮换通知后,生成(0,1)之间的随机数 $rand$,若 $rand<T(r)$,则向原 HN 发送功能节点申请,其中包含自己的 ID 和申请的功能节点类别;

(3) 若原 HN 收到超过预定数目的申请,则从收到的申请中随机选择新的 HN 和 KGN;若收到不足预定数目的申请包,则重复步骤(1)、步骤(2),直至选出功能节点,最后向新的功能节点发送确认包;

(4) 新 HN 和 KGN 收到确认包后确认自己新的功能节点身份;

(5) 新旧 HN 之间、新旧 KGN 之间交换信息,完成功能交替;

(6) 最后,由旧的 HN 通知簇内所有节点新 HN 的 ID.

4.5 EEHS在应急状态下的恢复功能

EEHS 在应急状态下的恢复功能是指网络遭受节点捕获攻击后的密钥系统和网络功能恢复,针对不同类的攻击可将恢复功能分为 4 类:

(1) 针对未形成共谋的 SN 节点捕获攻击的恢复.由于未形成共谋,因此可以利用那些未被捕获的密钥去更新被捕获的密钥,这也是基于 EBS 的动态密钥管理方法的一个重要优点;

(2) 针对形成共谋的 SN 节点捕获攻击的恢复.形成共谋后,可以利用共谋恢复密钥,最大限度地在未被捕获节点之间重新建立密钥体系,恢复网络功能;

(3) 针对 KGN 节点被捕获的恢复.KGN 被捕获后,存储在其上的管理密钥将被捕获.若并非所有 KGN 被捕获,则可以采用与第 1 类中类似的办法恢复密钥体系;若全部 KGN 均被捕获,则共谋已经形成,可以采用第 2 类中的恢复方法重建 EBS 体系;

(4) 针对 HN 节点被捕获的恢复.HN 被捕获是一种很危险的情况,因为簇内的很多数据业务和密钥事件都由 HN 控制,应立即驱逐被捕获的 HN,选取新的 HN 接替它,并重新建立密钥体系.

4.5.1 针对未形成共谋的 SN 节点捕获攻击的恢复

假设网络中节点 SN_a 被捕获,可以通过 HN 节点广播最多 m 个数据包更新被分配给 SN_a 的全部 k 个管理密钥 $K_a^{p_j}$,$j=1,\ldots,k,p_j\in\{1,2,\ldots,k+m\}$.这 m 个数据包为 $K_a^{q_l}(K_a^{p_1}((K_a^{p_1})'),K_a^{p_2}((K_a^{p_2})'),...,K_a^{p_k}(K_a^{p_k})')$,$l=1,2,\ldots,m$,$\{q_1,q_2,\ldots,q_m\}=\{1,2,\ldots,k+m\}-\{p_1,p_2,\ldots,p_k\}$,$(K_a^{p_j})'$ 为新的管理密钥.其中,$K_a^{p_j}((K_a^{p_j})')$ 由生成 $K_a^{p_j}$ 的 KGN 生成,最后再由生成 $K_a^{q_l}$ 的 KGN 进行加密,HN 只负责在簇内广播这些包.被驱逐的节点 SN_a 由于没有被分配 $K_a^{q_l}$,所以无法解密任何一个数据包,也就无法得到更新的密钥,因此被驱逐出了网络.而其他的节点拥有 $K_a^{q_l}$ 中的一个或几个,可以解密数据包,得到需要更新的管理密钥.显然,若先后有 y 个节点被捕获,则可以逐一地按照上面的方法驱逐它们.当 y 个节点被同时捕获但没有形成共谋时,可以证明通过广播最多 m^y 个数据包同时驱逐 y 个节点[11].恢复 EBS 体系后,可以按照第 4.3.4 节和第 4.3.5 节的方式恢复会话密钥和共谋恢复密钥.

4.5.2 针对形成共谋的 SN 节点捕获攻击的恢复

基于 EBS 的动态密钥管理方法中存在共谋问题,即对被捕获节点通过共享各自信息实施的联合攻击抵抗性较差,这是影响其安全性的主要因素,会使整个密钥体系被破坏.

设形成共谋的 y 个被捕获节点为 $SN_1,SN_2,\ldots,SN_y$,它们在初始化阶段得到的分层跳数 H_c 和分配给这些节点的共谋恢复密钥 K_{cr} 分别为 $H_c^1,H_c^2,...,H_c^y$ 和 $K_{cr}^1,K_{cr}^2,...,K_{cr}^y$,而 EEHS 中可以利用未被捕获的共谋恢复密钥去恢复与被捕获的 y 个节点具有不同 H_c 的节点,即利用集合 $\kappa=\{K_{cr}^i\mid K_{cr}^i\notin\{K_{cr}^1,K_{cr}^2,...,K_{cr}^y\}\}$ 中的密钥去恢复集合 $\Re=\{SN_i\mid H_c^i\notin\{H_c^1,H_c^2,...,H_c^y\}\}$ 中的节点.最好的情况是,y 个节点全部在同一个分层跳数 $\bar{H}_c$ 中,即 $\bar{H}_c=H_c^1=H_c^2=...=H_c^y$,显然 $1\leqslant\bar{H}_c\leqslant HTS$.若假设节点分布是均匀的,则可恢复率(可重建安全网络的 SN 占全部 SN 的比例) $\frac{(HTS-1)^2}{HTS^2}\leqslant R\leqslant\frac{HTS^2-1}{HTS^2}$;最坏的情况下,$y$ 个节点分布在全部分层跳数中,则 $\kappa=\varnothing$,$\Re=\varnothing$,因此可以恢复的节点比例 $R=0$.

具体的恢复过程如下:

(1) HN 通知 DT 发生共谋攻击,并从 DT 得到 K_{cr};

(2) HN 通知 $\Re$ 中的节点,发生共谋攻击,需要进行网络恢复;

(3) $\Re$ 中的节点重新向 HN 注册,HN 向 DT 进行注册,注册过程可参考第 4.4.1 节;

(4) HN 重新在 $\Re$ 中建立密钥体系,包括建立 EBS、选择 KGN、分配密钥等步骤,具体可参考第 4.4.2 节~第 4.4.5 节.

4.5.3 针对 KGN 节点被捕获的恢复

KGN 节点被捕获后的恢复工作由两部分组成——恢复密钥系统和重新选取 KGN.首先进行密钥系统恢复,当被捕获的 KGN 节点之间没有形成共谋时,由于每个 KGN 节点生成的密钥不超过 k 个,因此可以利用与第 4.5.1 节中相同的方法恢复密钥体系;若被捕获的 KGN 已经形成共谋,恢复过程将采用第 4.5.2 节中的方法.恢复密钥系统后,可以在恢复的网络中重新选举新的 KGN 取代旧的 KGN,最后完成全部恢复过程.先进行密钥恢复后选取 KGN,可以避免新 KGN 生成的新管理密钥泄露现象的发生.

4.5.4 针对 HN 节点被捕获的恢复

HN 节点掌握的网络内信息包括所有 SN 节点的 ID、KGN 节点的 ID 和 EBS 矩阵,因此这些信息均需要更新,这个过程需要 DT 的参与.具体的恢复过程如下:

(1) DT 通过私有密钥 K 更新被捕获 HN 所在簇内所有节点的 ID 和函数 F;

(2) DT 选择新的 HN;

(3) 新 HN 按照第 4.3 节的过程重新建立 EBS 密钥体系.

5 仿真与性能分析

基于 EBS 的动态密钥管理方法的性能评价主要由网络的抗攻击性能、能量有效性和节点所需密钥存储空间这 3 个方面决定.我们将本文提出的 EEHS 方法与 3 种经典的基于 EBS 的动态密钥管理方法 SHELL[9],SECK[11]和 LOCK[5]进行了比较.除此之外,还对 EEHS 中的分簇特性及参数选择进行了研究.仿真实验的网络拓扑结构如图 1 所示,在一个 600m×600m 的区域内随机布置 500 个节点,节点通信半径为 20m,DT 处于监测区域之外,在位置(800,300)处.

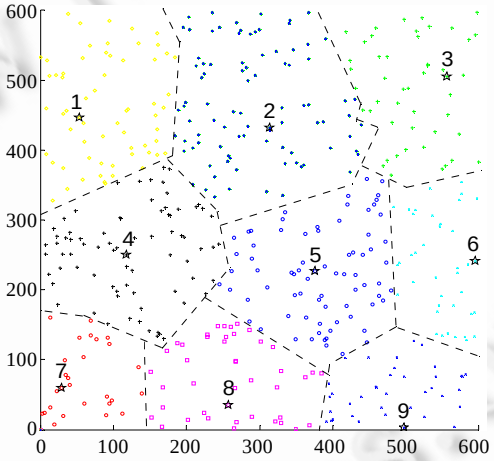

Fig.1 Network topology for the simulation and clustering results in EEHS

图 1 仿真实验采用的网络拓扑结构和 EEHS 的分簇结果

5.1 分簇特性与参数选择

在 EEHS 的分簇方法中,参数 p_{ini} 和 HTS 会直接影响分簇特性,如分簇过程耗时、簇的个数、簇内平均节点数目等.图 2 为分簇过程耗时及簇的个数与 p_{ini} 和 HTS 之间的关系曲线(p_{ini} 采用了对数坐标).显然,随着 p_{ini} 的减小,分簇所需的循环轮次会有所增加,而簇的个数会减少;同时从图 2 中还可以发现,当 p_{ini} 足够小时,例如 $p_{ini}<10^{-3}$ 时,簇的个数接近于一个只与 HTS 相关的稳定值.图 3 为在 p_{ini} 确定的情况下,簇的个数及簇内平均 SN 数目与 HTS 之间的关系曲线.簇的数目随着 HTS 的增加而减少,而簇内平均 SN 数目随着 HTS 的增加而增加.因此,在 p_{ini} 和 HTS 参数的选择上可以设定一个较小的 p_{ini},而通过调节 HTS 来调整簇的个数及簇的规模.图 1

是在 $p_{ini}=10^{-4}$,$HTS=10$ 条件下的分簇仿真结果,整个网络中的 500 个节点被分为了如图所示的 9 个簇.设定一个较小的 p_{ini} 会增加分簇过程的耗时,但并不会对 EEHS 的性能造成很大的影响,甚至是可以忽略不计的.因为,首先如第 4.2 节所述,分簇所需轮次并不是随着 p_{ini} 而呈线性增加的,而是近似与其倒数的对数成正比(如图 2 所示);其次,EEHS 中分簇过程只在网络布置的最初进行一次,而后分簇结构是不变的.

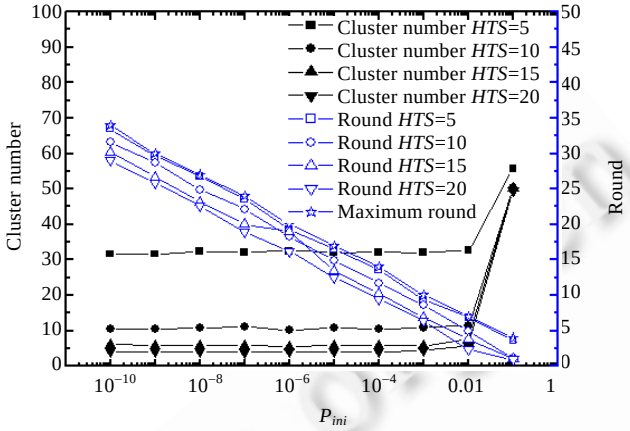

Fig.2 Relationship among cluster number, clustering rounds, p_{ini} and HTS

图 2 簇的数目、分簇轮次与 p_{ini},HTS 的关系曲线

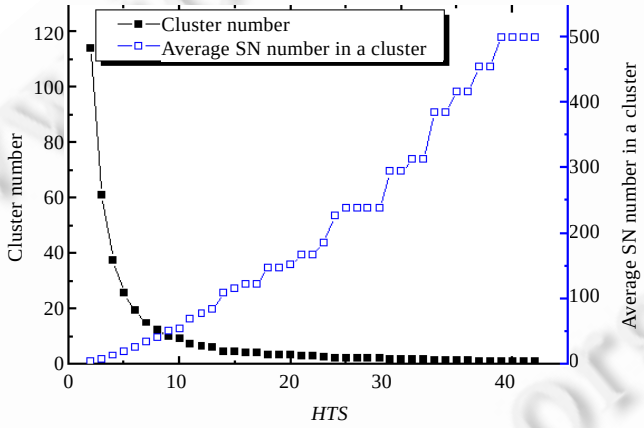

Fig.3 Relationship among cluster number, average SN number and HTS

图 3 簇的数目、簇内平均 SN 节点数和 HTS 的关系曲线

5.2 性能比较

LOCK,SECK,SHELL 与 EEHS 的分簇方法不同,这 3 种方法在节点布置之前需要预先选定簇的个数和簇头节点,并假设簇头节点的能耗无需考虑.因此,为保证 4 种方法具有可比性,尤其是在能耗方面,仿真实验中 EEHS 采用的是如图 1 所示的分簇结果,而其他 3 种方法则预设簇头节点为 9 个并随机布置在监测区域内.

5.2.1 网络的抗攻击性能

(1) 非共谋 SN 节点攻击的抵抗性能

图 4 为在 $k=3$,$m=5$(图 1 中簇 3 的 EBS 结构)条件下,4 种方法中被破解管理密钥比例与被捕获节点数的关系曲线.LOCK 中 t 的取值与 EEHS 不同,是为了保证它们在同等密钥存储空间占有量的情况下进行安全性比较.从图中我们可以发现,SECK 与 SHELL 的抗捕获性能最差,SHELL 稍强于 SECK.这是因为这两种方法只采用了

普通密钥.LOCK 由于采用了多项式密钥,所以抗捕获性能相比于 SHELL 和 SECK 得到了显著提高.EEHS 的被破解密钥比例又低于 LOCK,因为 LOCK 中的普通多项式密钥是 t-安全的,而 EEHS 中的同化多项式密钥是 t^2-安全的.图 5 为 EEHS 破解管理密钥比例与被捕获节点数的关系曲线随 t 的变化情况.由图可知,随着 t 的增加,EEHS 的安全性得到了显著的提高;尤其是当 $t\geqslant 5$ 时,被破解的管理密钥比例近似为 0.但与此同时,付出的代价是存储空间占有量变大.

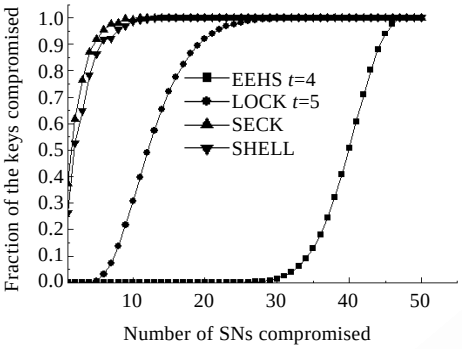

Fig.4 Relationship between the fraction of the keys compromised and the number of SNs compromised

图 4 破解管理密钥与被捕获节点数的关系曲线

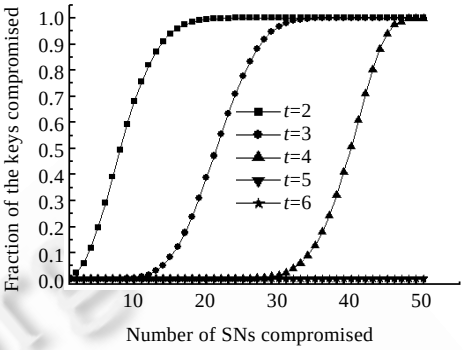

Fig.5 Relationship between the fraction of the keys compromised and the number of SNs compromised under the condition of different t

图 5 不同 t 下破解管理密钥与被捕获节点数的关系

(2) 共谋 SN 节点攻击的抵抗性能

当共谋形成时,SHELL 和 LOCK 的可恢复率为 0.SECK 按照 HN 与 SN 节点之间的相对方位将 SN 分类,并针对不同类的 SN 使用不同的共谋恢复密钥 K_{cr} 实现共谋后的恢复功能;而 EEHS 与 SECK 的区别在于,EEHS 是按照距离 HN 的跳数的不同对 SN 进行分类.图 6 是 4 种方法的恢复率与参与共谋的节点数目的关系曲线.

(3) HN 节点攻击的抵抗性能

LOCK 与 SHELL 中并未考虑 HN 被捕获后的恢复措施,所以一旦 HN 被捕获,整个簇将全部失效.EEHS 中采用功能节点轮换的方式,可以安全地驱逐被捕获的 HN 节点,并将簇内全部 SN 重新组网,所以恢复率可以达到 100%.SECK 中在节点布置阶段通过 Location Training 方式可以使 SN 节点保留备用 HN 节点信息,当主 HN 被捕获后,部分 SN 可以加入备用簇.图 7 为 4 种方法的恢复率与被捕获的 HN 数目之间的关系曲线(图中总结点数未包含被俘获节点).

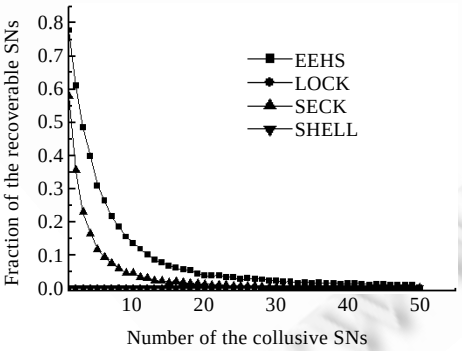

Fig.6 Relationship between the fraction of recoverable SNs and number of collusive SNs

图 6 共谋下网络可恢复率与共谋节点数目的关系

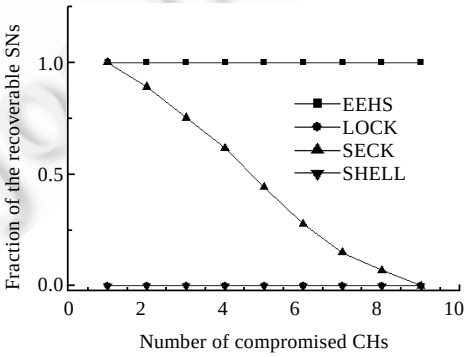

Fig.7 Relationship between the fraction of recoverable SNs and number of compromised CHs

图 7 网络可恢复率与被捕获 CH 节点数目的关系

5.2.2 能量有效性与网络寿命

降低能耗、提高能量有效性是无线传感器网络研究中的重点.但与传统的网络不同,无线传感器网络通常是作为一个整体来完成某些特定的任务,因此很多情况下,相比于节点的平均能耗,网络寿命更能反映网络的能量有效性.网络寿命有很多种表达方式[12],本文采用N-of-N形式的网络寿命,即网络的运行寿命到第1个节点出现能量枯竭时终止.我们利用NS2对4种方法的网络运行情况进行了仿真,簇内的通信采用最小跳数的路由方式和802.11协议,并设定节点的传感数据流量为1个数据包/秒,更新密钥频率为1次/1000秒,驱逐节点的频率为1个节点/3600秒,EEHS中功能节点轮换的时间间隔为1 000s,所有节点的初始能量均为10J.图8是4种方法的网络寿命比较,EEHS的网络寿命明显优于其他3种方法,这是因为EEHS中采用了功能节点轮换机制,平均了节点能耗延长了网络寿命.

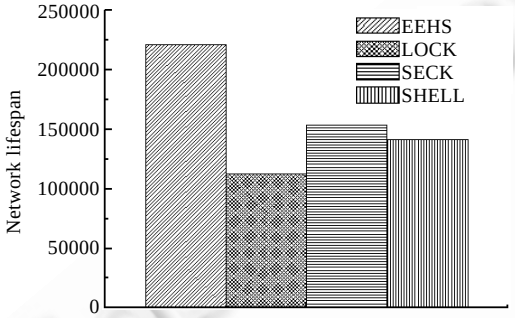

Fig.8 Network lifespan

图8 网络寿命

5.2.3 密钥空间

同为*EBS*(*n*,*k*,*m*),由于SHELL和SECK是基于普通密钥的,所以需要存储的密钥个数为*k*;而EEHS和LOCK是基于多项式的,所以它们需要存储的密钥个数分别为*k*(*t*+2)和*k*(*t*+1)个.虽然EEHS需要的存储空间相比于其他方法变大,但是相比于静态密钥管理方法所需的存储空间(如文献[3])仍是少量的.例如,当取EBS结构*EBS*(50,3,5)时,图9为节点需存储密钥数目比较情况.其中,文献[3]是在保证连通度为0.999,且破解全部密钥所需节点个数和EEHS相同的条件下进行比较的.

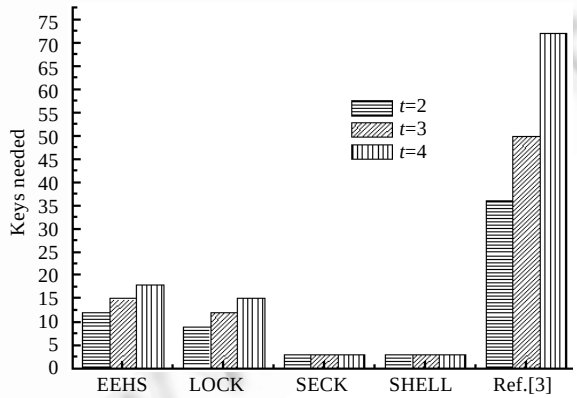

Fig.9 Memory space for keys

图9 密钥存储空间

5.2.4 计算资源开销

EEHS与利用普通多项式的EBS都采用基于多项式的密钥,其密钥计算资源开销为相应的多项式求值的复

杂度.多项式求值的方法有很多,其适用性和复杂度也各不相同,本文利用最简单的直接求值的方法对 EEHS 和利用普通多项式的 EBS 这两种密钥方案进行计算复杂度分析.

直接计算 a_ix^i 需要 i 次乘法运算,利用普通多项式的 EBS 中的密钥是通过计算形如 $\sum_{i=0}^{t}a_ix^i$ 的一元 t 次多项式得到的,因此需要 $\frac{t(t+1)}{2}$ 次乘法运算和 t 次加法运算.EEHS 中,密钥计算分为两步:

(1) 利用节点序号数组(ID_1,ID_2),通过同化三元多项式 $f(x_1,x_2,x_3)=C+\sum_{i_1=1}^{t+1}\sum_{i_2=1}^{t+1}\sum_{i_3=1}^{t+1}a_{i_1i_2i_3}x_1^{i_1}x_2^{i_2}(x_3-x_c)^{i_3}$ 得到形如 $C+\sum_{i=1}^{t+1}a_i(x-x_c)^i$ 的一元 t 次多项式;

(2) 利用 x_c,通过步骤(1)中得到的一元 t 次多项式得到密钥.

步骤(1)中需要$(t+1)^2(t+2)$次乘法运算和$(t+1)^3$ 次加法运算;步骤(2)中需要 $\frac{(t+1)(t+2)}{2}$ 次乘法运算和 $t+2$ 次加法运算.实际上,EEHS 最终的计算复杂度并不是两个步骤的简单相加,因为只有在同化三元多项式被捕获的时候,步骤(1)才会发生,而步骤(2)是常用的密钥计算过程,如密钥更新等.因此,本文方法的密钥计算复杂度主要由步骤(2)所决定,而它又与利用普通多项式的 EBS 近似.

综合上面的仿真结果和分析可以发现,基于同化多项式密钥分享的 EEHS 优势主要体现在安全性方面,它可以有效而明显地提高网络的抗捕获性能.取得这一优势的同时虽然付出了一定的存储资源,但这只是与在存储空间方面性能非常优秀的 LOCK,SHELL 等相比,而与静态的密钥管理方法相比,所需存储资源仍是较少的.

6 结 论

本文提出了 EEHS——一种基于 EBS 的适合于大规模分簇式无线传感网络的动态密钥管理方法.EEHS 是一个全面而且安全的无线传感器网络密钥管理解决方案,它的主要组成部分及特点包括:(1) 一种 t^2-安全的特殊多项式密钥(同化多项式密钥)用于提高网络的抗捕获性能;(2) 一种不需要任何特殊节点或功能、分簇性能可调而且适合于EBS体系特点的分簇方法;(3) 一种安全的密钥体系建立和运行机制;(4) 功能节点轮换功能用于均衡节点能耗,提高网络鲁棒性;(5) 4 种针对不同危害程度攻击的网络功能恢复机制.仿真与分析结果表明,与相关文献相比,EEHS 可以显著地提高网络的抗捕获性能,还能够有效地延长网络的运行寿命.在存储空间占有量方面虽略有增加,但是相比于静态密钥管理方法所需的存储空间仍是少量的.

References:

[1] Ren FY, Huang HN, Lin C. Wireless sensor networks. Journal of Software, 2003,14(7):1282−1291 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/14/1282.htm

[2] Du WL, Deng J, Yunghsiang SH, Pramod KV. A key predistribution scheme for sensor networks using deployment knowledge. IEEE Trans. on Dependable and Secure Computing, 2006,3(1):62−77. [doi: 10.1109/TDSC.2006.2]

[3] Eschenauer L, Gligor VD. A key-management scheme for distributed sensor networks. In: Proc. of the 9th ACM Conf. on Computer and Communications Security. Washing: ACM Press, 2002. 41−47.

[4] Eltoweissy M, Heydari H, Morales L, Sadborough H. Combinatorial optimization of key management in group communications. Journal of Network and Systems Management, 2004,12(1):33−50. [doi: 10.1023/B:JONS.0000015697.38671.ec]

[5] Eltoweissy M, Moharrum M, Mukkamala R. Dynamic key management in sensor networks. IEEE Communications Magazine, 2006, 44(4):122−130. [doi: 10.1109/MCOM.2006.1632659]

[6] Riaz R, Ali A, Kim KH, Ahmad F, Suguri H. Secure dynamic key management for sensor networks. In: Proc. of the Innovations in Information Technology. Dubai: IEEE Press, 2006. 1−5.

[7] Moharrum M, Eltoweissy M, Mukkamala R. Dynamic combinatorial key management scheme for sensor networks. Wireless Communication and Mobile Computing, 2006,6(7):1017−1035. [doi: 10.1002/wcm.435]

[8] Li LC, Li JH, Tie L, Pan J. ACKDs: An authenticated combinatorial key distribution scheme for wireless sensor networks. In: Proc. of the 8th ACIS Int'l Conf. on Software Engineering, Artificial Intelligence, Networking, and Parallel Distributed Computing (SNPD). Qingdao: IEEE Press, 2007. 262−267.

[9] Younis MF, Ghumman K, Eltoweissy M. Location-Aware combinatorial key management scheme for clustered sensor networks. IEEE Trans. on Parallel and Distributed Systems, 2006,17(8):865−882. [doi: 10.1109/TPDS.2006.106]

[10] Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. on Wireless Communications, 2002,1(4):660−670. [doi: 10.1109/TWC.2002.804190]

[11] Chorzempa M, Par JM, Eltoweissy M. Key management for long-lived sensor networks in hostile environments. Computer Communications, 2007,30(3):1964−1979. [doi: 10.1016/j.comcom.2007.02.022]

[12] Kong FR, Li CW, Zhao XD, Ding QQ, Jiao F, Gu QB. An energy-efficient and low-latency sink positioning approach for wireless sensor networks. LNCS 4864, 2007. 123−134.

附中文参考文献:

[1] 任丰原,黄海宁,林闯.无线传感器网络.软件学报,2003,14(7):1282−1291. http://www.jos.org.cn/1000-9825/14/1282.htm

孔繁瑞(1981－),男,辽宁沈阳人,博士生,主要研究领域为无线传感器网络的安全性.

李春文(1958－),男,博士,教授,博士生导师,主要研究领域为复杂网络,非线性控制理论,电力系统优化.