doi:10.3969/j.issn.1000-436X.2012.10.013

WSN 中基于全同态加密的对偶密钥建立方案

张永[1]，温涛[1,2]，郭权[2]，李凤坤[2]
(1. 东北大学 软件中心，辽宁 沈阳 110004；2. 大连东软信息学院 计算机科学与技术系，辽宁 大连 116023)

摘 要：针对 Guo 等人基于排列的多对称多项式方案提出一种攻击方法，证明其方案未能突破容忍门限，并不能抵御大规模节点俘获攻击。通过引入全同态加密提出一种对偶密钥建立方案,使共享密钥计算过程在加密状态下完成，阻止了敌手获得与多项式有关的信息，成功应对了大规模节点俘获攻击。提出一种全同态加密体制的间接实现方法，降低了方案的存储及计算复杂度。分析及实验表明本方案的存储、计算和通信开销完全满足无线传感器网络的要求。
关键词：无线传感器网络；对偶密钥建立；同态加密；密钥管理
中图分类号：TP393 **文献标识码**：A **文章编号**：1000-436X(2012)10-0101-09

Pair-wise key establishment for wireless sensor networks based on fully homomorphic encryption

ZHANG Yong[1], WEN Tao[1,2], GUO Quan[2], LI Feng-kun[2]
(1. Software Center, Northeastern University, Shengyang 110004, China;
2. Department of Computer Science and Technology, Neusoft Information Institute, Dalian 116023, China)

Abstract: An attack on the permutation-based multi-polynomial scheme exposed in the paper of Guo was proposed.The scheme couldn't frustrate the large-scale node capture attack was proved. A pair-wise key establishment scheme was proposed by introducing homomorphic encryption thought, which was used to protect polynomials and made all keys be established in encrypted state. Therefore, the large-scale node capture attack was thwarted because adversaries couldn't get any information about polynomials from the encrypted data used to establish keys. A method was presented to achieve fully homomorphic encryption indirectly, which used much less storage and computation resource than existing fully homomorphic encryption. The analysis and experiment show our pair-wise key establishment scheme has very good performance in terms of storage, computation as well as communication and is suitable to wireless sensor networks with limited capability.
Key words: WSN; pair-wise key establishment; homomorphic encryption; key management

1 引言

无线传感器网络(WSN, wireless sensor network)因其广阔的应用前景而越来越受到人们的关注[1]。但由于无线传感器网络的特点，解决其安全问题是非常棘手的，因此，无线传感器网络的安全问题一直是学术界研究的热点。

密钥管理技术是保证网络安全的基石和核心技术，特别是对偶密钥建立(PKE, pair-wise key establishment)问题，因而得到了广泛的研究。自 2002 年，Eschenauer 和 Gligor 首次研究对偶密钥建立问题，并提出基于随机密钥池[3]的方案之后，虽然出现了大量的对偶密钥建立方案[2,4~7]，但是这些方案都无一例外地沿用着文献[3]最初设计的 3 个基本步骤：密钥预置、共享发现及路径密钥建立(path-key establishment)。因此，这些方案都可以归

收稿日期：2011-02-11；**修回日期**：2011-11-23
基金项目：国家自然科学基金资助项目 (61170168, 61170169)
Foundation Item: The National Natural Science Foundation of China (61170168, 61170169)

为一类，称为三步法方案。三步法方案存在的主要问题：①功能不足，没有很好地实现某些关键需求，特别像可扩展性、节点移动性、全连通性、抵御大规模节点攻击等需求；②必须建立路径密钥，需要很大的无线通信量，导致能耗很高，不符合低能耗要求。2007 年，Zhang 等在文献[8]中提出了引入扰动的多项式方案，第一次较好地实现了前述的关键需求，并且他们的方案仅需要很少的无线通信量，从而开启了对偶密钥建立方案的一个新类别——全功能方案。尽管在此之前基于文献[15]的对称多项式理论已经提出了很多方案，但是在那些方案中，多项式是在三步法的框架下作为一种密钥产生方法使用的。与之不同，Zhang 等的方案以多项式为主体，利用其对称性为任意 2 节点提供共享密钥，从而实现了前述的关键需求。但是，由于多项式存在容忍门限——能够容忍的可被俘获的最大节点数，不能自然地容忍大规模节点共谋或俘获攻击，因此，Zhang 等在文献[8]中的主要工作是谋求打破容忍门限的方法，以期成功应对大规模节点共谋攻击。遗憾的是，在 2009 年 Albrecht 等证明文献[8]引入扰动的方法并不能打破容忍门限，所以不能解决大规模节点共谋攻击问题[10]。鉴于全功能方案的优势，Guo 等在文献[9]中提出了基于排列的多项式方案，通过增加多项式重构难度，成功应对了基于 Lagrange 插值法的攻击，声称实现了全功能方案。而实际上，Guo 等的方案并不能解决大规模节点共谋攻击问题，这一点将在本文的第 3 节给出证明。所以，虽然学术界看好全功能方案的优势，但是至今还没有真正可行的方案。表 1 给出了现有 PKE 方案功能的实现情况。

表 1　　现有 PKE 方案功能实现的情况

方案	RLNCA	全连通	FDKE	节点移动性(动态拓扑)	可扩展性
三步方案[2~7]	\	×	×	×	×
准全功能方案[8,9]	×	√	√	√	√
本文的方案	√	√	√	√	√

说明：RLNCA(robustness to large-scale node capture attack)是指抵御大规模节点俘获攻击或共谋攻击的能力；全连通，是指网络中任意 2 个节点都可以建立共享密钥；完全直接密钥建立(FDKE, full direct key establishment)，是指保证所有共享密钥都是直接密钥、不存在间接密钥；“\”表示只有部分方案能够达标。

本文的贡献：①针对文献[9]基于排列的多对称多项式方案提出一种攻击方法，证明其方案未能突破容忍门限，不能抵御大规模节点共谋攻击；②通过引入同态加密思想保护多项式运算过程，提出一种打破容忍门限的方法，实现了一个真正可行的全功能方案；③提出一种全同态加密体制的间接实现方法，大大降低了现有全同态加密实现方法的存储及计算复杂度，从而使基于全同态加密的方案满足了传感器网络的资源要求。

2　预备内容

2.1　符号定义

文中用到的符号如表 2 所示。注意：本文认为大规模节点共谋攻击和大规模节点俘获攻击是同一概念。

表 2　　符号定义

符号	含义
$E_k(m)$	利用密钥 k 对消息 m 进行加密
$D_k(c)$	利用密钥 k 对密文 c 进行解密
$E_k^i(u)$	$E_k^i(u)=(E_k(u))^i$
mac(.)	为消息认证码运算
$m_1 \| m_2$	消息 m_1 和 m_2 进行连接

2.2　系统模型

本文方案基于的无线传感器网络模型由大规模的低能耗、低成本的传感器节点组成。节点具有有限的能量供给、存储空间和计算能力。各节点资源相当、功能对等，通过相互协作、自组织地完成网络功能。在这种网络模型中，没有基站式的中心控制设施，并且各节点是可移动的。假定存在一个离线的权力机构(OLA, off-line authority)。该 OLA 负责建立网络、配置节点启动信息、收集数据等操作，是网络所属实体(或单位)的抽象，不参与网络运行过程，不会导致单点失败。

2.3　全同态加密

同态加密[11](HE, homomorphic encryption)是由 Rivest 等在 1978 年提出的，简单地说，是指加密系统具有式(1)所示的加法属性或式(2)所示的乘法属性。

$$E_k(m_1)+E_k(m_2)=E_k(m_1+m_2) \tag{1}$$

$$E_k(m_1)\times E_k(m_2)=E_k(m_1\times m_2) \tag{2}$$

但是，2009 年以前，加密系统具有的同态性都是部分的，即式(1)或式(2)之一成立。2009 年，Gentry 首次提出了同时具备 2 个同态运算属性的加密系统——全同态加密(FHE, fully homomorphic encryption)系统[12]，方案基于的是理想格(ideal

lattices)。2010 年，Dijk 等发现基于初等模运算就可以实现全同态性，提出了一种更简单的构造方法[13]。但是，遗憾的是，为了保证安全强度，上述 2 种实现方法均需要很长的密钥长度，从而带来较大的存储开销和计算代价，因此，这 2 种实现方法都还未达到实用水平。

基于同态原理，数据可以在加密状态下进行运算或处理，对数据运算不再需要先进行解密操作，因而可以发展许多重要的应用。例如，机密数据的代理运算。从这个角度看，本文发展了全同态性的一个新应用方向——多项式加密，即加密多项式的运算过程。

2.4　二元对称多项式

1984 年，Blom 提出了基于对称矩阵的对偶密钥协商算法[14]。之后，Blundo 等基于这种思想提出了(BSP, bivariate symmetric polynomial)[15]。基本过程如下。

OLA 从有限域 GF(p)中(其中，p 为大质数)随机选取一组数$\{a_{i,j}\mid 0<i<t,0<j<t\}$，构成对称二元多项式，如式(3)所示，并使等式 $f(x,y)=f(y,x)$ 成立。

$$f(x,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{i,j}x^i y^j \tag{3}$$

ID 为 u 的节点在部署前需要预置一元多项式，如式(4)所示。

$$g_u(y)=f(u,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{i,j}u^i y^j=\sum_{j=0}^{t}c_j^u y^j \tag{4}$$

其中，$c_j^u=\sum_{i=9}^{t}a_{i,j}u^i$。

节点 u，v 的共享密钥 $K_{u,v}$ 的协商过程由式(5)保证。

$$g_u(v)=f(u,v)=f(v,u)=g_v(u) \tag{5}$$

文献[15]证明上述基于多项式的密钥协商算法是无条件抵御 t 个节点共谋攻击的，或者说，只要敌手俘获的节点数不超过 t，那么，敌手不可能得到关于共享密钥的任何信息。然而，为了使方案能够抵御大规模节点的俘获攻击，需要提高方案的容忍能力，很多方案都失败了[10]，多项式的阶 t 又变成了方案容忍能力的门限。

定义 1　在基于 t 阶多项式的方案中，多项式自身能够容忍的可被俘获的最大节点数是 t，称 t 为容忍门限。

3　对基于排列的多对称多项式方案的攻击

为了解决基于多项式方案中存在的容忍门限问题，文献[9]引入基于排列的多个对称多项式方案，并声称打破了容忍门限。其基本思想是在敌手使用插值法求解多项式的过程中引入困难性。下面首先简单介绍一下该方案，并说明其所谓的困难性，然后给出本文的攻击方法。

3.1　基于排列的多对称多项式方案

为了便于理解本文的攻击方法，这一节简单介绍一下与理解攻击方法有关的内容。至于方案的详细内容请参照文献[9]。

文献[9]的方案可以分为 2 步：预置多项式和建立共享密钥。预置多项式包括以下 3 步。

1) OLA 随机选择 m 个二元对称多项式，$f_i(x,y)(1\leqslant i\leqslant m)$。

2) 为每一个节点分配 ID $u\in\mathrm{GF}(p)$，并构造 m 个一元多项式，如式(6)所示。式中的“[i]”表示多项式序号，以区别指数运算。

$$g_u^{[i]}(v)=f_i(u,y),\ 1\leqslant i\leqslant m \tag{6}$$

3）将 2)步中产生的 m 个一元多项式以任意顺序预置到节点 u 中，如式(7)所示。

$$g_u^{[i_1]},g_u^{[i_2]},\cdots,g_u^{[i_m]} \tag{7}$$

其中，序列$\langle i_1,i_2,\cdots,i_m\rangle$是序列$\langle 1,2,\cdots,m\rangle$的一个排列。

预置结束、节点部署后，节点间开始建立共享密钥。以 u, v 为例介绍该过程如下。

1) 2 节点交换 ID，节点 u 得到 IDv;

2) 节点 u 将 v 代入自己的多项式序列，得

$$g_u^{[i_1]}(v),g_u^{[i_2]}(v),\cdots,g_u^{[i_m]}(v) \tag{8}$$

3）计算式(9)得共享密钥 $K_{u,v}$。

$$K_{u,v}=g_u^{[i_1]}(v)\oplus g_u^{[i_2]}(v)\oplus\cdots\oplus g_u^{[i_m]}(v) \tag{9}$$

文献[9]认为敌手只有通过 Lagrange 插值法(称为插值攻击)才能突破上述过程。过程如下。

1) 俘获 t+1 个节点，提取每个节点中的 m 个一元多项式并记为一组，共有 t+1 个组。

2) 对 t+1 个组进行重新分组，使得来自同一个二元对称多项式的一元多项式组成一个新组，共有 m 个新组，每个新组含有 t+1 个一元多项式。

3) 采用 Lagrange 插值法，处理每个新组，可求得 m 个二元对称多项式。

由 Lagrange 插值法可知，对于一个 t 阶多项式，只要求得 t+1 个解，便可以成功求得该多项式。由式(7)可知，多项式在预置到节点上时，顺序已经全部打乱，即每个节点上多项式的顺序是任意的。因此，攻击方法的第 2)步是不确定的，需要尝试不同的分组方法。重新分组的过程是一个排列组合问题，其组合数如式(10)所示。由式(10)可知该问题的计算复杂度是 $O(m!)$，因此是一个 NP 问题。这就是文献[9]所谓的困难性。文献[9]认为利用该困难突破了容忍门限，可以容忍大规模节点的俘获攻击。

$$com_m^k = (m!)^k \tag{10}$$

3.2 攻击方法

为了清楚地说明问题，给出下面的定义 2。

设网络 $N=\{N_1,\cdots,N_n\}$，n 为节点总数，$F=[F_1,\cdots,F_k]\subset N$ 为俘获的节点集合，$P(N_i)$ 表示节点 N_i 上预置的一元多项式或一元多项式集合。

定义 2 设 A 为一种针对基于多项式对偶密钥建立方案的攻击方法，如果在 A 的作用下，$\forall n\in N-F$，能够求得 $P(n)$，那么，称 A 攻破了整个网络的密钥建立方案。

利用 Lagrange 插值法求解式(3)所示的二元多项式 $f(x,y)$，然后就可以求得每个节点上如式(4)所示的一元多项式，从而能够攻破整个网络的密钥建立方案，这是最直接的方法。由 3.1 节的分析可知，文献[9]在求二元多项式 $f(x,y)$的过程中引入了如式(10)所示的组合困难，成功应对了基于 Lagrange 插值法的攻击，以下简称插值攻击。但是，求出所有节点上的一元多项式，或者说，攻破整个网络的密钥建立方案不必一定要求出二元多项式 $f(x,y)$。下面将给出不通过求解二元多项式 $f(x,y)$，而直接求解某一未被俘获节点上的一元多项式的方法，具体方法如下。

本文的攻击方法基于文献[16]中 2.2 节提出的中间模型及多项式构造方法。基本过程是寻求一个多项式集合 $f_1\cdots f_m$ 来完全描述一个包含 m 个映射关系的黑盒 $B=(B_1(x),\cdots,B_m(x))$，即求多项式集合 $f_1(x)\cdots f_m(x)$ 使得对于任意的 x 有集合 $(B_1(x),\cdots,B_m(x))$ 与集合 $\{f_1(x)\cdots f_m(x)\}$ 相等[16]。

攻击过程如下。

设已经俘获的 mt+1 个节点组成集合 $F=\{F_1,\cdots,F_{mt+1}\}$，其中，$m$ 为节点上预置的多项式个数，t 为预置的多项式阶数，“$\lfloor m \rfloor$”表示整数集合{1,2,⋯,m}。设将要攻破的节点 ID 为 u ($u\notin F$)，即要求出节点 u 上的一元 t 阶多项式集合 $\left\langle g_u^{[1]}(x),g_u^{[2]}(x),\cdots,g_u^{[m]}(x)\right\rangle$。

1) 将 u 分别代入 F 中的每一个节点，得式(11)所示的实例集合。由多项式的对称性可知，式(11)所示的实例集合可以由节点 u 上的多项式集合求得，但是，需要将 $F=\{F_1,\cdots,F_{mt+1}\}$ 代入节点 u 的多项式集合。换言之，式(11)所示集合与式(12)所示的集合相等。因此，可以构造节点 u 上多项式集合的 mt+1 个实例值，如式(13)所示。

$$\begin{aligned}&\{(y_1,y_2,\cdots,y_m)\ |\\ &y_1=g_{F_i}^{[1]}(u),y_2=g_{F_i}^{[2]}(u),\cdots,y_m=g_{F_i}^{[m]}(u),\\ &1\leqslant i\leqslant mt+1\}\end{aligned} \tag{11}$$

$$\begin{aligned}&\{(y_1,y_2,\cdots,y_m)\ |\\ &y_1=g_u^{[1]}(F_i),y_2=g_u^{[2]}(F_i),\cdots,y_m=g_u^{[m]}(F_i),\\ &1\leqslant i\leqslant mt+1\}\end{aligned} \tag{12}$$

$$\begin{aligned}&\{(x;y_1,y_2,\cdots,y_m)\ |\\ &x=F_i,y_1=g_u^{[1]}(F_i),y_2=g_u^{[2]}(F_i),\cdots,\\ &y_m=g_u^{[m]}(F_i),\\ &1\leqslant i\leqslant mt+1\}\end{aligned} \tag{13}$$

$$e_j(x)=\sum_{S\subset\lfloor m\rfloor,|S|=j}\prod_{i\in S}g_u^{[i]}(x) \tag{14}$$

2) 利用式(13)所示的集合对式(14)进行插值运算，以求得 x^d 的系数 $e_{j,d}$。

3) 利用 2)步求得的各系数构造二元多项式(15)。

$$Q(x,y)=\sum_{j=0}^{m}\sum_{d=0}^{ji}(-1)^i e_{j,d}x^d y^{m-j} \tag{15}$$

4) 对 $Q(x,y)$进行二元因式分解得式(16)。

$$Q(x,y)=\prod_{i=1}^{m}(y-f_i(x)) \tag{16}$$

式(16)产生的多项式集合 $\{f_1(x)\cdots f_m(x)\}$ 即为所求的 $\left\langle g_u^{[1]}(x),g_u^{[2]}(x),\cdots,g_u^{[m]}(x)\right\rangle$。

上述攻击过程的复杂度为 $O(m\log m\log\log m)$，并没有像文献[9]预期的攻击方法那样存在 NP 问题。本文称上述针对黑盒模型构造多项式的方法为黑盒构造法，并称基于该构造法的攻击方法为黑盒攻击法。

4 对偶密钥建立方案

由第 3 节可知，由 Guo 等提出的基于排列的多对称多项式全功能方案不能抵御大规模节点俘获攻击，因而也是不安全的，所以，现有的全功能方

案都是不安全、不可行的。下面将给出本文的全功能对偶密钥建立方案。其基本思想是：通过同态加密技术保护多项式运算过程，使共享密钥计算过程在加密状态下进行，从而使敌手即使俘获了节点也不能得到关于多项式的信息，进而不能利用插值攻击法或黑盒攻击方法重构多项式，最终实现抵御大规模节点俘获攻击的目的。方案的具体内容如下。

4.1　准备工作

节点部署前，OLA 需要做以下 2 步全局的初始化工作。

1) 为每一个节点分配一个唯一的 IDu ($u \in$ GF(p))，并随机选择如式(3)所示的对称多项式，2 个变量的阶都是 t。

2) OLA 选择一种全同态加密方法，例如，文献[12]或文献[13]的方案，并生成自己的公钥 PK_{OLA}。把私钥 SK_{OLA} 丢弃。

4.2　秘密信息预置

完成 4.1 节的准备工作后，OLA 需要对每个节点做如下的预置工作。设节点 ID 为 u。

1) 将节点 u 代入式(3)，产生式(4)所示的一元多项式。

2) 利用 PK_{OLA} 对式(4)所示的一元多项式进行加密。首先，对节点 IDu 进行如式(17)所示的计算，得 u 的密文，称为 u 的秘密身份；其次，求式(3)系数 a_{ij} 的密文 $E_{PK_{\mathrm{OLA}}}(a_{ij})(1\leqslant i\leqslant t,1\leqslant j\leqslant t)$；最后，按式 $c_j^u=\sum_{i=0}^{t}a_{i,j}u^i$ 计算 $c_j^u(1\leqslant j\leqslant t)$ 的加密值 $E_{PK_{\mathrm{OLA}}}(c_j^u)$, 其中, $1\leqslant j\leqslant t$，构造式(4)的加密形式，如式(18)所示。

$$E_{PK_{\mathrm{OLA}}}(x)=E_{PK_{\mathrm{OLA}}}(u) \tag{17}$$

$$\begin{aligned}&E\Big(\sum_{i=0}^{t}\sum_{j=0}^{t}a_{i,j}u^iy^j\Big)\\&=\sum_{i=0}^{t}\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(a_{ij})E^i_{PK_{\mathrm{OLA}}}(u)E^i_{PK_{\mathrm{OLA}}}(y)\\&=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^u)E^i_{PK_{\mathrm{OLA}}}(y)\end{aligned} \tag{18}$$

3) 将 $E_{PK_{\mathrm{OLA}}}(c_j^u)$ (其中，$1\leqslant j\leqslant t$)、$E_{PK_{\mathrm{OLA}}}(u)$ 和 PK_{OLA} 预置到节点 u 中。特别注意：将 $E^i_{PK_{\mathrm{OLA}}}(y)$ 整体看作一元多项式的变量。

4.3　密钥建立

下面介绍部署后节点 u 与相邻的节点 v 建立共享密钥的过程。节点 u 构造如式(19)所示的消息。

$$\left\langle\begin{matrix}\mathrm{hello},u,E_{PK_{\mathrm{OLA}}}(u) & | \\ mac(\mathrm{hello},u,E_{PK_{\mathrm{OLA}}}(u))\end{matrix}\right\rangle \tag{19}$$

节点 v 收到消息后，进行消息完整性验证。如果不完整，则丢弃；如果完整，则取出 $E_{PK_{\mathrm{OLA}}}(u)$，代入式(18)所示的一元多项式得密钥 $K_{v,u}$，如式(20)所示，并构造如式(21)所示的回复消息。

$$\begin{aligned}K_{v,u}&=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^v)E^i_{PK_{\mathrm{OLA}}}(y)\\&=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^v)E^i_{PK_{\mathrm{OLA}}}(u)\end{aligned} \tag{20}$$

$$\left\langle\begin{matrix}\mathrm{ok},v,E_{PK_{\mathrm{OLA}}}(v) & | \\ mac(\mathrm{ok},v,E_{PK_{\mathrm{OLA}}}(v))\end{matrix}\right\rangle \tag{21}$$

节点 u 收到回复消息后，进行消息完整性验证。如果不完整，则丢弃；如果完整，则取出 $E_{PK_{\mathrm{OLA}}}(v)$，代入式(18)得密钥 $K_{u,v}$，如式(22)所示。

$$\begin{aligned}K_{u,v}&=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^u)E^i_{PK_{\mathrm{OLA}}}(y)\\&=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^u)E^i_{PK_{\mathrm{OLA}}}(v)\end{aligned} \tag{22}$$

上述过程完成后，节点 u 和节点 v 建立起共享密钥 $K_{u,v}=K_{v,u}$。

5　安全性分析

5.1　对偶密钥生成方法的正确性

定理 1　4.3 节给出的共享密钥建立过程是正确的，或者说 $K_{u,v}=K_{v,u}$。

证明　由式(22)可知

$$K_{u,v}=\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(c_j^u)E^i_{PK_{\mathrm{OLA}}}(v)$$

将式(18)代入上式得

$$K_{u,v}=\sum_{i=0}^{t}\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(a_{ij})E^i_{PK_{\mathrm{OLA}}}(u)E^i_{PK_{\mathrm{OLA}}}(v)$$

由全同构加密的性质得

$$\begin{aligned}K_{u,v}&=\sum_{i=0}^{t}\sum_{j=0}^{t}E_{PK_{\mathrm{OLA}}}(a_{ij}u^iv^j)\\&=E_{PK_{\mathrm{OLA}}}\Big(\sum_{i=0}^{t}\sum_{j=0}^{t}(a_{ij}u^iv^j)\Big)\\&=E_{PK_{\mathrm{OLA}}}(f(u,v))\end{aligned}$$

同理，可得

$$K_{v,u}=\sum_{i=0}^{t}\sum_{j=0}^{t}E_{PK_{OLA}}(a_{ij}v^{i}u^{j})$$
$$=E_{PK_{OLA}}(\sum_{i=0}^{t}\sum_{j=0}^{t}(a_{ij}v^{i}u^{j}))$$
$$=E_{PK_{OLA}}(f(v,u))$$

又由 $f(x,y)$ 的对称性可知

$$f(u,v)=f(v,u)$$

所以，得

$$K_{u,v}=E_{PK_{OLA}}(f(u,v))=E_{PK_{OLA}}(f(v,u))=K_{v,u}$$

证毕。

定理 1 表明任意 2 个节点都可以在交换秘密身份之后，通过计算协商出共享密钥。

5.2 大规模节点俘获攻击的安全性

下面分析本文的方案抵御大规模节点俘获攻击的能力。无论是插值攻击法还是黑盒攻击法，都需要获取一定数量的实例值才能重构多项式。本文的方案有定理 2 所述的性质。

定理 2 本文的方案能够有效地阻止敌手获取多项式的实例值。

证明 由 4.2 节的 3)步可知，预置到每个节点的一元多项式都是加密的，即如式(18)所示。又由 4.3 节的密钥建立过程可知，欲建立共享密钥的两节点首先需要交换各自的秘密身份，然后在同构原理的支持下进行共享密钥的计算，最终双方计算获得共享密钥。整个过程可以抽象为式(23)。

$$E_k(f(x,y)) \tag{23}$$

由式(23)可知，敌手俘获节点后得到的是加密状态的多项式，由于没有密钥，因而敌手无法得到多项式，进而无法得到多项式的实例值。而且还可以看出，无论敌手俘获多少节点，都不能改变这种情况。证毕。

由于无法获得有效的实例值，因此，有定理 3 的结论。

定理 3 本文的方案能够抵御插值攻击和黑盒攻击。

证明 由定理 2 的结论可知，本文的方案能够有效地阻止敌手获取多项式的实例值，又因为插值攻击和黑盒攻击都是依赖实例值集合的，所以，由于不能获取实例值，2 种攻击方法都是无效的。证毕。

黑盒攻击法对于基于多项式的方案来说是一种很强的攻击，原因是其基于黑盒模型，重构多项式集合的数据是集合类型的，不要求有顺序，甚至还允许有少量的噪声。所以，无论对多项式引入扰动[8]还是打乱多项式的顺序[9]都无法阻止敌手在俘获一定量的节点后成功构造多项式实例值。由前面的分析可知，把每个节点上的多项式看作黑盒中的关系，由于本文的方案阻止了实例值的获取，从而有效地抵御了黑盒攻击。但是，如果把加密状态的多项式(式(23))看作黑盒中的关系，那么本文的方案还安全吗？下面进行分析。

定理 4 式(23)所示的加密状态的多项式关系不再是简单的多项式关系。

证明 反证法。设式(23)所示的加密状态的多项式关系是简单的多项式关系，那么可以通过插值法或黑盒法求得所有密文服从的多项式，从而找出明文与密文之间的关系，换言之，使用的同态加密方案是不安全的。而本文选择的全同态加密方案(文献[12]或文献[13])是安全的，产生矛盾。所以，式(23)所示的加密状态的多项式关系不再是简单的多项式关系。

定理 4 的证明过程表明，如果采用安全的全同态加密方案，那么，式(23)所示加密后的多项式关系不再是简单的多项式关系。由文献[16]可知，黑盒构造法成功的前提是盒内的关系是多项式关系。如果盒内的关系并非多项式关系，那么利用黑盒构造法无法成功描述黑盒。所以，本文的方案还是安全的。

6 功能及性能分析

6.1 本文方案的全功能性

实际上，进行简单的分析就可以看出，方案的全连通性[8,9,15]、完全直接密钥共享、移动性及可扩展性等都是基于对称多项式的方案自然支持的。因此，唯一需要解决也是最难解决的是大规模节点俘获攻击。下面进行分析。

从过程抽象表达式(23)整体上看，由于 $\forall u,v(u=v\Rightarrow E_k(f(u,v))=E_k(f(v,u)))$，所以，加密后的多项式仍然具有对称性质，因此，从网络中任意取 2 节点，都可以建立起对偶密钥，所以，网络是全连通的。同样的理由，因为任意 2 个节点都可以建立对偶密钥，所以，建立的所有对偶密钥都是直接的，不存在路径密钥，因而是完全的直接密钥。由于方案没有利用、也不依靠任何网络拓扑信息或节点位置信息，所以，方案支持节点移动。由 4.2 节秘密信息预置过程可知，将节点 ID 代入二元对称多项

式就可以得到惟一的一个一元多项式，再对该一元多项式加密就可以装配一个新节点，因此，对节点数量没有任何限制，也没有任何父代、子代的区别和限制。所以，方案具有可扩展性。由 5.2 节可知，方案具有抵御大规模节点俘获攻击的能力。综上所述，本文的方案是一种全功能方案。

6.2　全同态加密体制的间接实现方法

第 4 节给出了基于全同态加密的对偶密钥建立方案，但是，由于现有的全同态加密方法都还未达到实际应用水平，因而，还需要寻求全同态加密体制的实现方法。在这一节中，本文并不是给出一种新的全同态加密构造方案，而是寻求一种计算式的方法。

基本思想是：利用加法代替乘法的技术将全同态过程嫁接到同态加密方案(或称半同态加密方案)上，换言之，使用同态加密方案实现全同态过程。修改第 4 节的有关步骤如下。

1) 将 4.1 节准备工作的第 2)步修改为：OLA 选择一种具有加法同态性的加密方案，例如 Paillier[17]、Benaloh[18]等，并生成自己的公钥 PK_{OLA}。把私钥 SK_{OLA} 丢弃。

2) 将 4.2 节的第 2)步修改为：对式(4)所示的一元多项式进行加密。首先，对式(4)所示的一元多项式进行改造，如式(24)所示。

$$g_u(y)=f(u,y)=\sum_{i=0}^{t}\sum_{j=0}^{t}a_{i,j}u^i y^i=\sum_{j=0}^{t}c_j^u y^j=\sum_{j=0}^{t}\underbrace{c_j^u+c_j^u+\cdots+c_j^u}_{y^j} \tag{24}$$

加密后如式(25)所示。

$$E_{PK_{\mathrm{OLA}}}\Big(\sum_{i=0}^{t}\sum_{j=0}^{t}a_{i,j}u^i y^i\Big)=\sum_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^u)+\cdots+E_{PK_{\mathrm{OLA}}}(c_j^u)}_{y^j} \tag{25}$$

3) 将 4.2 节的第(3)步修改为：将 $E_{PK_{\mathrm{OLA}}}(c_j^u)$ 和 PK_{OLA} (其中，$0\leqslant j\leqslant t$)预置到节点 u 中。

4) 在 4.3 节中，节点进行密钥协商时，发送的消息改为式(26)和式(27)。

$$\langle \mathrm{hello},u\,|\,mac(\mathrm{hello},u)\rangle \tag{26}$$

$$\langle \mathrm{ok},v\,|\,mac(\mathrm{ok},v)\rangle \tag{27}$$

5) 各节点收到消息后按式(28)或式(29)所示的过程进行共享密钥的计算。

$$K_{u,v}=\sum_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^u)+\cdots+E_{PK_{\mathrm{OLA}}}(c_j^u)}_{y^j}=\sum_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^u)+\cdots+E_{PK_{\mathrm{OLA}}}(c_j^u)}_{v^j} \tag{28}$$

$$K_{v,u}=\sum_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^v)+\cdots+E_{PK_{\mathrm{OLA}}}(c_j^v)}_{y^j}=\sum_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^u)+\cdots+E_{PK_{\mathrm{OLA}}}(c_j^u)}_{u^j} \tag{29}$$

第 4 节给出的全同态方案和本节的改造方案本质是相同的。不同的是改造方案基于同态加密，在计算上是可行的。此外，改造后方案在消息大小、预置的秘密信息大小等方面性能还有所提升。具体的比较如表 3 所示。

表 3　　2 种方案的比较

比较项目	全同态方案	改造方案
同态性	全同态	(半)同态
计算复杂度	高	低
消息内容	包括 ID 和加密的 ID	仅包括 ID
预置的秘密	$E_{PK_{\mathrm{OLA}}}(c_j^u)$、$E_{PK_{\mathrm{OLA}}}(u)$ 和	$E_{PK_{\mathrm{OLA}}}(c_j^u)$ 和
($0\leqslant j\leqslant t$)	PK_{OLA}	PK_{OLA}

6.3　存储空间分析

设所有的参数及密钥都在有限域 $GF(p)$上，并记其长度为 l(byte)。

由 6.2 节的改造方案可知，本方案在部署前要求节点预置的数据包括 2 部分：加密后的多项式和 OLA 的公钥。假设使用的同态加密体制是 Paillier。则存储占用量 S 可表示为式(30)，其中，t 是指多项式的阶数。

$$S=(t+3)l \tag{30}$$

由式(30)可知，存储空间是 t 的函数。由 5.2 节可知，本方案抵御大规模节点俘获攻击依赖的是具有同态性质的加密体制，换言之，能否抵御大规模节点俘获攻击与 t 的取值关系不大，不再像其他基于多项式的方案那样，t 与方案的安全级别是耦合的。因此，t 取值不必太大，所以，方案的存储占用率很小。

6.4　通信量分析

由于每种传感器产生相同无线通信量所用的能耗并不同，所以本文在进行通信能耗分析时，以通信量为单位。通信量用消息数量和大小来衡量。

方案无线通信量的大小与节点装配的协议栈有关。如果节点装配了像 ZigBee 这样的协议栈，由于 ZigBee 的网络层具有邻居发现机制，所以，方案直接利用该功能提供的节点信息进行共享密钥的计算即可，无需再进行额外的无线通信。如果节点所装配的协议栈不具备邻居发现机制，那么需要进行无线通信以确定与哪一节点建立共享密钥。但是，由于需要交换的信息仅是节点 ID，并且每个节点为建立一个共享密钥仅发送一个消息，所以，所需的无线通信量是十分有限的。例如，ZigBee 规定 ID 为 16bit 的整数。

6.5 计算量分析

虽然本文的方案涉及到公钥加密体制，但是，由第 4 节全同态方案和 6.2 节的改造方案可以看出，在方案启动后，没有任何加、解密操作发生，代价最高的运算形式来源于加法同态性运算过程。以 Paillier 为例，分析如下。

由 Paillier 同态性(如式(31)所示)，可知，用加密的系数计算共享密钥 $K_{u,v}$ 需要进行如式(32)所示的运算(以节点 u 计算与节点 v 的共享密钥为例)。

$$E_k(a)E_k(b)=E_k(a+b) \tag{31}$$

$$K_{u,v}=\prod_{j=0}^{t}\underbrace{E_{PK_{\mathrm{OLA}}}(c_j^u)\times\cdots\times E_{PK_{\mathrm{OLA}}}(c_j^u)}_{v^j} \tag{32}$$

由式(32)可知，原操作为乘法，其频度如式(33)所示。所以，算法的渐近复杂度为

$$O(v^t)=v+v^2+\cdots+v^t \tag{33}$$

由 $O(v^t)$ 可知，算法的渐近复杂度与节点的 ID 和多项式的阶数两者有关。但是，由 Zigbee 规范可知，节点 ID 一般为 16bit 整数，所以，最坏的情况为 $O(65\,535^t)$。而又由 6.3 节的分析可知，t 取值不必太大，一般取 3，所以，复杂度为 $O(v^t)$ 的算法是传感器节点完全可以负担的。假设网络规模为最大值 65 536，并且任意 2 节点进行通信的概率相等，则由分析可知，最坏情况出现的概率如式(34)所示，约为十万分之一。其中，N 表示网络节点总数。

$$N-1/C_N^2=65\,535/C_{65\,536}^2\approx 3/10^5 \tag{34}$$

7 实验

本节将对方案的计算性能进行实验，以测试方案在实际平台上的表现。实验以 TI 公司的 CC2 430 为平台，以 IAR 嵌入式 IDE 为开发环境，采用汇编语言进行开发。

CC2 430 实验平台的配置情况如表 4 所示。

表 4 实验用 CC2 430 配置

项	配置	项	配置
CPU	8 051	时钟频率	32M
Flash	128KB	RF	2.4GHz
RAM	8KB		

结果如图 1 与图 2 所示。图 1 展示了节点建立对偶密钥需要的时间。图 2 展示了相应的能耗变化。

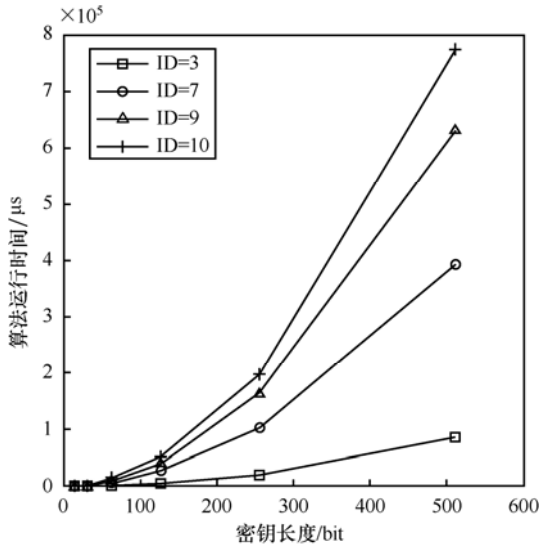

图 1 算法运行时间随密钥长度的变化

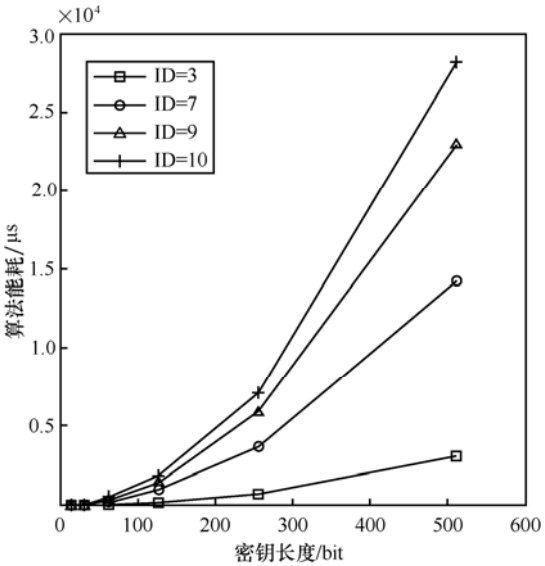

图 2 能耗随密钥长度的变化

由图 1 与图 2 分析可知，方案建立对偶密钥的能耗量级为毫瓦级，符合无线传感器网络的能

耗要求。

8　结束语

在本文中，首先利用基于文献[16]构造的黑盒攻击对文献[9]的方案进行了攻击，证明文献[9]的方案不能抵御大规模节点攻击，不是安全的全功能方案，因此，现有的全功能对偶密钥建立方案都不能抵御大规模节点俘获攻击，都是不安全的。鉴于全功能方案的优势，本文在深入研究了基于多项式的方案存在的容忍门限问题之后，引入同态加密思想对多项式的运算过程进行保护，从而成功应对了黑盒攻击，突破了容忍门限，解决了大规模节点俘获攻击问题。此外，为了降低全同态加密体制的存储及计算复杂度，本文还提出了一种全同态加密体制的间接实现方法。性能分析认为采用间接实现方法后的对偶密钥建立方案使用的存储空间和无线通信量十分有限，计算复杂度虽然为 $O(v^t)$，但由于 v 和 t 的值都比较小，传感器节点完全可以负担。综上所述，本文首次实现了一个真正可行的全功能对偶密钥建立方案。

参考文献：

[1] XIAO Y, RAYI V K, SUN B, *et al*. A survey of key management schemes in wireless sensor networks[J]. Computer Communications, 2007, 30(11-12):2314-2341.

[2] LIU D G, NING P. Location-based pairwise key establishment for static sensor networks[A]. Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks[C]. Fairfax, Virginia, USA, 2003.72-82.

[3] ESCHENAUER L, GLIGOR V. A key management scheme for distributed sensor networks[A]. The 9th ACM Conference on Computer and Communication Security[C]. Washington D C, USA, 2002. 41-47.

[4] CHAN H W, PERRIG A, SONG D. Random key predistribution schemes for sensor networks[A]. Proceedings 2003 Symposium on Security and Privacy[C]. Carnegie Mellon, PA, USA, 2003.197-213.

[5] LIU D, NING P. Establishing pairwise keys in distributed sensor networks[A]. The 10th ACM Conference on Computer and Communications Security[C]. Washington D C, USA, 2003.52-61.

[6] MARTIN K, PATERSON M. Key predis -tribution for homogeneous wireless sensor networks with group deployment of nodes[J]. ACM Transactions on Sensor Networks, 2010,7(2):1-18.

[7] LIU D, NING P, DU W. Group-based key pre-distribution in wireless sensor networks[J]. ACM Transactions on Sensor Networks, 2008, 4(2):11-20

[8] ZHANG W, TRAN M, ZHU S, *et al*. A random perturbation-based scheme for pairwise key establishment in sensor networks[A]. Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing[C]. New York, NY, USA, 2007. 90-99.

[9] GUO S, LEUNG V, QIAN Z Z. A permutation-based multi-polynomial scheme for pairwise key establishment in sensor networks[A]. IEEE International Conference on Communications (ICC)[C]. Cape Town, South Africa, 2010.1-5.

[10] ALBRECHT C, HALEVI G, KATZ J. Attacking cryptographic schemes based on perturbation polynomials[A]. ACM Conference on Computer and Communication Security[C]. Chicago, IL, USA, 2009.1-8.

[11] RIVEST R, SHARMIR A, DERTOUZONS M.　On Data Banks and Privacy Homomorphisms[M]. Orlando: Academic Press, 1978.

[12] GENTRY C. Fully homomorphic encryption using ideal lattices[A]. Proceedings of the 41st Annual ACM Symposium on Theory of Computing[C]. New York, USA, 2009.169-178.

[13] DIJK M, GENTRY C, HALEVI S, *et al.*　Fully homomorphic encryption over the integers[A]. Cryptology-EUROCRYPT[C]. French Riviera, Springer, 2010.24-43.

[14] BLOM R. An optimal class of symmetric key generation systems[A]. Cryptology[C]. Santa Barbara, California, USA, 1984. 335-338.

[15] BLUNDO C, SANTIS A D, HERZBERG A, *et al*. Perfectly-secure key distribution for dynamic conferences[A].Advances in Cryptology-CRYPTO'92, LNCS740[C]. Springer, German, 1993.471-486.

[16] AR S, LIPTON R, RUBINFELD R, *et al*. Reconstructing algebraic functions from mixed data[J]. SIAM Journal on Computing, 1998, 28(2): 487-510.

[17] PAILLIER P. Public-key cryptosystems based on composite degree residuosity classes[A]. Advances in Cryptology-EUROCRYPT'99[C]. Prague, Czech Republic, 1999. 223-238.

[18] BENELOH J. Dense probabilistic encryption[A]. Proceedings of the Workshop on Selected Areas of Cryptography[C]. Kingston, England, 1994. 120-128.

[19] ZIGBEE A. Zigbee specification[EB/OL]. http://www.daintree.net/whatsnew/070130-spec.php, 2006.

作者简介：

张永（1981-），男，山东莱芜人，东北大学博士生，主要研究方向为无线网络安全。

温涛（1962-），男，陕西宝鸡人，博士，东北大学教授、博士生导师，主要研究方向为网络安全、知识组织。

郭权（1973-），男，辽宁大连人，博士，大连东软信息学院教授，主要研究方向为计算机网络。

李凤坤（1983-），女，山东青岛人，硕士，大连东软信息学院讲师，主要研究方向为网络安全。