WSN 中同构模型下动态组密钥管理方案

温涛[1,2]，张永[1]，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：研究了同构网络模型的组密钥管理问题，首次给出了一个明确的、更完整的动态组密钥管理模型，并提出了一种基于多个对称多项式的动态组密钥管理方案。该方案能够为任意多于 2 个且不大于节点总数的节点组成的动态多播组提供密钥管理功能，解决了多播组建立、节点加入、退出等所引发的与组密钥相关的问题。该方案支持节点移动，具有可扩展性，并很好地解决了密钥更新过程中多播通信的不可靠性。组成员节点通过计算获得组密钥，只需要少量的无线通信开销，大大降低了协商组密钥的代价。分析比较认为，方案在存储、计算和通信开销方面具有很好的性能，更适用于资源受限的无线传感器网络。

关键词：无线传感器；同构网络模型；动态多播组；组密钥管理
中图分类号：TP393 **文献标识码**：A **文章编号**：1000-436X(2012)06-0164-10

Dynamic group key management scheme for homogeneous wireless sensor networks

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

Abstract: Focuses on the study of GKM scheme applicable to homogenous network model， proposed a clear and complete key management model for dynamic groups for the first time and exposed a dynamic group key management (DGKM) scheme based on symmetric polynomials. The scheme provided a solution to related group key management issues, such as key establishment, key updating, node addition and node revocation, for a multicast group consisting of any number (bigger than 2 and less than the total number) of nodes. Besides, the scheme supported node mobility as well as scalability and coped well with the unreliable wireless multicast communication. Group members can get group key by computing and with little wireless communication, therefore, greatly reduced the cost of key agreement. The analysis shows the scheme has very good performance in terms of storage, computation as well as communication and is more suitable to wireless sensor networks with limited capability.

Key words: wireless sensor networks; homogeneous network model; dynamic multicast group; group key management

1 引言

无线传感器网络(WSN, wireless sensor networks)因其广阔的应用前景而越来越受到人们的关注[1]。但由于其受限的资源、无安全保障的工作环境和低成本要求等原因[2]，安全性问题一直是学术界研究的热点。

密钥管理是一种基本的安全机制。密钥管理方案[3,4]包括偶性密钥管理(PKM, pair-wise key management)和组密钥管理(GKM, group-wise key management)。现有文献在讨论 PKM 时，大多基于分布式网络模型[3]或同构网络模型[5,6]——节点功能

收稿日期：2010-12-22；修回日期：2011-03-20

对等、没有中心控制设施(例如基站)，关注网络自组织性，认为同构网络模型比异构网络模型更具有一般性，也更符合低成本的要求；而在讨论 GKM 时，大多基于异构网络模型——分层分组的层簇式结构[7,8]或中心式结构(具有密钥服务器[9]、组密钥管理器[10]或基站)，对同构模型 GKM 讨论很少。基于异构网络模型的 GKM 在创建多播组时需要中心控制点，因此，无法直接将基于异构模型开发的方案直接移植到同构网络中。

一定数量的节点根据需要动态形成多播组，节点协作与计算都通过多播组进行，从而将协作的规模限定在一定范围内，这种协作称为有界协作。有界协作对能耗控制有重要意义，因而得到广泛应用。网内处理 (in-network processing) 技术就利用了这种有界协作，例如数据聚集。组内成员通过多播通信启动聚集要比通过单播高效，显示出发展动态多播组的重要性[5]，从而也使得解决其安全性问题更加紧迫。

与异构网络模型的组密钥管理相比，同构模型的组密钥管理发展滞后，密钥管理模型不完整、不明确。文献[11]首先讨论了动态会议（或组）密钥分发问题，提出了基于对称多项式的密钥分发方案，着力解决了组的动态性——一定数量的任意节点可以形成组。但是文献只是从数学角度分析了提出的 2 种密钥分发方法在适应组或会议动态性方面的优势，并不是一个完整的组密钥管理方案。文献[12]讨论了在没有中心控制设施（例如基站或 KS 等）条件下的组密钥管理问题。首先利用节点的初始可信性——网络部署后最初的一段时间内所有节点都是可信的——构造安全的局部链接，然后通过安全的局部链接实现其他的安全管理功能，例如组密钥更新。但是，该方案基于的组密钥管理模型不完整，还有很多问题没有解决(详见第 2.2 节)。于是，本文研究了同构网络下动态组密钥管理机制。

本文应用 Blundo 等人提出的基于 k-顽健 t-组对称多项式的密钥管理方案来解决同构网络模型的组密钥管理问题，主要解决了 3 个关键问题。①组密钥更新问题。由文献[11]可知，多项式份额在节点部署前被预置到节点中，节点部署后，任意 t 个节点可以协商一个组密钥。但是，相同的 t 个节点组成的组只能协商一个密钥。组密钥是静态的，无法更新。如果密钥应用环境的生命周期较短，这种情况是允许的，例如文献[11]提到的动态会议，生命周期较短。如果密钥应用环境的生命周期很长，那么密钥无法更新是严重的安全隐患。因此，这是一个必须解决的问题。但是，多项式份额已经预置，并且节点也已经部署，因此，无法修改多项式，所以，这也是个比较棘手的问题。②组成员数量不变的问题。多播通信中，组成员的数量是变化的，或者说是不确定的。而文献[11]的密钥协商过程要求每个组的成员数量必须是 t，多于或少于 t 都无法协商出组密钥。所以，需要解决组成员数量不变的问题。同样，这也是一个很难解决的问题。③应对大规模节点共谋攻击问题。无线传感器网络部署在无人管理的环境下，俘获大量节点在很多情况下并不困难[13]，而文献[11]仅能容忍至多 k 个节点的共谋攻击。

本文第 2 节介绍了系统模型、组密钥管理模型及本方案基于的 k-顽健 t-组多项式等内容；第 3 节给出了密钥管理方案的具体过程；第 4 节分析了 3 个关键问题的解决方法；第 5 节给出了安全性分析及与文献[12]的比较结果；第 6 节分析了方案的存储、计算和通信开销及与文献[12]性能比较的结果；第 7 节总结了本文的工作。

2　预备内容

2.1　系统模型

本方案基于的无线传感器网络模型由低能耗、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。

本文假设如下。

1) 节点具有多播功能，但不要求多播是可靠的，即承认多播消息可能不会到达全部的节点。

2) 假定存在一个离线的权力机构(OLA, off-line authority)负责建立网络、配置节点启动信息、收集数据等操作。该机构是网络所属实体(或单位)的抽象，不参与网络运行过程，不会导致单点失败。

3) 节点具有识别恶意节点的能力。在前述的网络模型中，所有的网络功能都需要各个节点协作完成，因此，节点应该具有识别恶意节点的能力。例如节点具有入侵检测[14]功能。

2.2　同构网络组密钥管理模型

目前，同构网络的组密钥管理与异构网络的相

比，管理模型不明确。例如，文献[12]把整个网络看作一个组，没有组标识，无法根据网络需求动态组建组大小可变的多播组。下面给出一个明确的、更完整的同构网络组密钥管理模型应该支持的内容。

1) 动态性。任意多于 2 个且不大于节点总数的节点组成的多播组的密钥管理，或称动态组的密钥管理。

2) 多角色。任意节点可以参与多个多播组；任意节点可以发起多个多播组；发起多播组的节点是该组的管理者。组管理者具有管理密钥更新、节点加入、退出及组销毁等权限。

3) 成员关系可变更性。节点可加入或退出多播组。

4) 移动性。支持节点移动。

5) 健壮性。容忍多播不可靠的能力。

6) 可扩展性。支持大规模网络及追加节点的能力。

该多播模型的核心思想是任何节点都可以根据网络业务需要自行组织多播组，成为组管理者，从而要求其他节点协作以完成自己的业务。

2.3 *k*-顽健 *t*-组对称多项式

从有限域 GF(*p*)中(其中 *p* 为大质数)随机选取一组数$\{a_{i_1,\cdots,i_t} \mid 0<i_1<k,\cdots,0<i_t<k\mid\}$，构成对称多元多项式。

$$f(x_1,\cdots,x_t)=\sum_{i_1=0}^{k}\cdots\sum_{i_t=0}^{k}(a_{i_1,\cdots,i_t}(x1)^{i_1}\cdots(xt)^{i_t}) \quad (1)$$

使等式$f(x_1,\cdots,x_t)=f(x_{\alpha_1},\cdots x_{\alpha_t})$成立，其中，序列$\langle\alpha_1,\cdots,\alpha_t\rangle$为序列$\langle1,\cdots,t\rangle$的任意排列。从文献[11]的定理 4.2 可知，具有 *t* 个 *k* 阶变量的对称多项式能够为具有 *t* 个成员的组分配一个共享密钥，并且能够容忍 *k* 个任意节点的共谋攻击。本文称具有这种能力的对称多项式为 *k*-顽健 *t*-组对称多项式。

2.4 虚拟节点和组密钥变更戳

定义 1 组密钥管理方案认为在网络中存在只有 ID，但并没有对应物理实体的节点，称这类节点为虚拟节点。

本文的密钥管理方案对虚拟节点设定了下面的规则。

规则 1 虚拟节点不能预置式(3)所示的多项式序列。

规则 2 没有虚拟节点的多播组不能加入新节点。

规则 1 的原因是虚拟节点不是真实的节点，无法预置多项式序列；规则 2 的原因是没有虚拟节点的多播组成员数已经达到最大量 *t*。

定义 2 组密钥变更戳(GKCS, group key-changing stamp)是指满足下面条件的函数$\pi=f(\cdot)$：

1) 自变量是组成员共知的内容，该内容没有机密性要求，但要求严格的同步。

2) 设自变量 x_1 和 x_2，如果$x_1\neq x_2$，则$f(x_1)\neq f(x_2)$。

3 组密钥管理方案

组密钥管理方案主要包括节点部署前的准备工作及 4 个基本的组密钥管理动作：组密钥建立、密钥更新及节点加入和强制退出。

3.1 准备工作

OLA 需要做以下的准备工作。

首先，对节点数量 *n* 进行估计，*n* 等于当前部署的节点数量与将来各次追加的数量之和。选择一大质数 *p*($p>n$)，有限域 GF(*p*)。

其次，OLA 随机选择 *m* 个 *k*-顽健 *t*-组对称多项式$f_j(x_1,\cdots,x_t)$，$1\leqslant j\leqslant m$，每一个多项式具有 *t* 个变量，每个变量的阶都是 *k*，且$\forall j$，有等式$f_j(x_1,\cdots,x_t)=f_j(x_{\alpha_1},\cdots,x_{\alpha_t})$成立，其中，序列$\langle\alpha_1,\cdots,\alpha_t\rangle$为序列$\langle1,\cdots,t\rangle$的任意排列。

变量 *t* 表示组成员的最大量。变量 *t* 的值由 OLA 根据网络需求决定。变量 *m* 和 *k* 的值将在 5.2 节讨论。

最后，OLA 对节点进行下面的预置[13]。

1) 为每一个节点分配一个 ID，φ ($\varphi\in$ GF(*p*))，并预置一个单向函数 *h*(·)，其运算结果为 GF(*p*)内的值。

2) 设置 *t*−2 个虚拟节点 (virtual nodes)，它们 ID 分别为$v_1=1,\ v_2=2,\cdots,v_{t-2}=t-2$。

3) 计算下面 *m* 个多项式：

$$g_j^{\varphi}(x_2,\cdots,x_t)=f_j(\varphi,x_2\cdots,x_t) \quad (2)$$

其中，$1\leqslant j\leqslant m$，$\varphi\notin\{1,2,\cdots,(t-3),(t-2)\}$。

4) 将步骤 3)中产生的 *m* 个多项式以任意顺序预置到节点φ中[13]，即有如下多项式序列：

$$\left\langle g_{j_1}^{\varphi},\ g_{j_2}^{\varphi},\cdots,g_{j_m}^{\varphi}\right\rangle \quad (3)$$

其中，序列($\langle j_1,j_2,\cdots,j_t\rangle$)为序列$\langle1,2,\cdots,t\rangle$的任意排

列。

3.2　动态组密钥建立

节点部署后，设节点 λ 要发起多播组 ω，该组由 $\delta(\delta \leqslant t)$ 个成员节点组成，$\omega=\{\varphi_1,\cdots,\varphi_\delta\}$。节点 λ 生成多播消息如式(4)所示。

$$\left\langle \begin{matrix} hello, h(\xi), \omega, \lambda \quad | \\ mac(hello, h(\xi), \omega, \lambda) \end{matrix} \right\rangle \tag{4}$$

其中，$h(\xi)$ 为密钥变更戳；ξ 为组密钥更新秘密，由节点 λ 秘密保存，在下一次组密钥更新时使用；“|”为消息连接运算符；$mac(\cdot)$为消息认证码运算。

ω 中节点收到信息后按下面的过程计算多播组 ω 的共享密钥 K_ω。设 $\varphi_i \in \omega$，其计算 K_ω 的过程如下。

1) 验证消息完整性，如果通过，向下进行；否则，丢弃，退出处理。

2) 提取发起者 ID，判断自己是否信任该节点。如果不信任，丢弃，退出处理；如果信任，向下进行。

3) 保存 $h(\xi)$ 后，构造节点 ID 集合 $v_1, v_2, \cdots, v_{t-\delta-1}, \lambda, \varphi_1 \cdots \varphi_{i-1}, \varphi_{i+1}, \cdots, \varphi_\delta$，并分别代入 m 个多项式得式(5)。由多项式的对称性可知，节点代入顺序不影响最终结果。

4) 将上一步中计算的各多项式的值代入式(6)，计算得出 K_ω 的值。

$$\begin{aligned} q_{j_1} &= g_{j_1}^{\varphi_i}(v_1,\cdots,v_{t-\delta-1},\lambda,\varphi_1,\cdots\varphi_{i-1},\varphi_{i+1}\cdots\varphi_\delta) \\ q_{j_2} &= g_{j_2}^{\varphi_i}(v_1,\cdots,v_{t-\delta-1},\lambda,\varphi_1,\cdots\varphi_{i-1},\varphi_{i+1}\cdots\varphi_\delta) \\ &\vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ q_{j_m} &= g_{j_m}^{\varphi_i}(v_1,\cdots,v_{t-\delta-1},\lambda,\varphi_1,\cdots\varphi_{i-1},\varphi_{i+1}\cdots\varphi_\delta) \end{aligned} \tag{5}$$

$$K_\omega = K_{\varphi_i} = h(h(\xi) \wedge q_{j_1} \wedge q_{j_2} \cdots \wedge q_{j_m}) \tag{6}$$

其中，“$\wedge$”表示做二进制与运算。需要说明的是，这里的“$\wedge$”还可以替换为“$\vee$”或“$\oplus$”(二进制或运算或异或运算)。

5) 按式(7)计算组标识。

$$ID_\omega = h(\omega \bigcup \lambda) = h(\lambda, \varphi_1, \cdots \varphi_\delta) \tag{7}$$

3.3　密钥更新

节点 λ 产生新的组密钥更新秘密 ξ'，产生多播消息：

$$\left\langle \begin{matrix} update, ID_\omega, \xi, h(\xi') \quad | \\ mac(update, ID_\omega, \xi, h(\xi')) \end{matrix} \right\rangle \tag{8}$$

设 $\varphi_i \in \omega$，其计算新组密钥 K'_ω 的过程如下：①验证消息完整性，如果通过，向下进行；否则，丢弃，退出处理。②取出 ξ，做 $h(\xi)$ 运算，并与保存的值进行比较，如果相等，认为消息来源于组管理者 λ，否则，丢弃，退出处理。③保存 $h(\xi')$，并代入式(6)得新密钥 K'_ω。

3.4　节点加入组或强制节点退出

当有新节点需要加入多播组或节点 λ 识别出多播组中存在恶意节点时，它产生新的组密钥更新秘密 ξ'，新的组成员列表，生成多播消息：

$$\left\langle \begin{matrix} member, \xi, h(\xi')\ \omega' \quad | \\ mac(member, \xi, h(\xi')\ \omega') \end{matrix} \right\rangle \tag{9}$$

设 $\varphi_i \in \omega$，其计算新组密钥 K'_ω 的过程如下：①验证消息完整性，如果通过，向下进行；否则，丢弃，退出处理。②取出 ξ，做 $h(\xi)$ 运算，并与保存的值进行比较，如果相等，则认为消息来源于组管理者 λ；否则，丢弃，退出处理。③保存 $h(\xi')$。根据 ω' 构造新的节点 ID 集合，代入式(5)产生新的 q 值集合。将 $h(\xi')$ 和新的 q 值集合代入式(6)，得新密钥 K'_ω。④计算新的组标识 $ID'_\omega = h(\omega')$。

4　关键问题

4.1　组密钥更新问题

现有的基于对称多项式的方案(如文献[11]和文献[13])都没有解决密钥更新问题。难点是多项式份额(PS, polynomial share)已经预置到节点并且节点已经部署，因此无法变更。

本文的解决思想是：如果组密钥产生过程像文献[11,13]提出的方案那样只依赖于 PS，不可能产生出新的共享密钥。如果再预置 m 个多项式，无疑增加了存储开销，并且也只能再产生一个新的共享密钥，没有解决问题。因此，只有在建立组密钥时，引入另外的信息，拓展组密钥空间，才可能解决问题。

本文引入组密钥变更戳的概念，如 3.2 节式(6)中的 $h(\xi)$，标识密钥变更，拓展了组密钥空间。最终使得组密钥更新的次数不受限制，解决了密钥更新问题。

此外，由 $h(\cdot)$函数的单向性可知，即使敌手破解了组密钥，也无法由组密钥推出集合 $\{q_{j_1}, q_{j_2}, \cdots, q_{j_m}\}$。因此，采用新的密钥变更戳，就又可

以建立新的组密钥。

4.2 组成员数量不变的问题

由对称多项式计算过程可知，基于对称多项式的密钥管理方案[5,11,13]，都要求每个多播组的成员数量必须是 t(文献[5]中，t=2)，否则无法计算多项式，也就无法建立组密钥。

本文的解决方法是引入虚拟节点。由于虚拟节点的特殊性——只有节点 ID，而网络中并不存在真实的节点，可以帮助成员数不足 t 的组建立组密钥。从而打破了多项式对组成员数量的限制，解决了第 2 个关键问题。

由 3.2 节可知，当组的成员数量不足 t 时，由虚拟节点(从 v_1=1 开始)补足 t。虽然从多播组的角度看，成员个数不足 t 个(虚拟节点不是真的节点)，但是从多项式的角度看，成员个数为 t 个(虚拟节点具有 ID)，可以形成一个可计算的组。

4.3 应对大规模节点共谋攻击问题

应对大规模节点共谋攻击问题的解决，本文借鉴并发展了文献[13]基于排列的方案。发展的方面是：①引入密钥变更戳，并采用单向函数对集合 $\{q_{j_1}, q_{j_2}, \cdots, q_{j_m}\}$ 进一步保护。②式(6)中采用的运算可扩展为“或”、“与”或者“异或”运算，增加了选择的空间。基于多个对称多项式排列的安全性在文献[13]中有详细的分析。为了本文的完整性，将在 5.2 节中从多项式求解的角度再进行简单的分析。

4.4 处理多播通信的不可靠性

由 4.1 节~4.3 节可知，密钥建立、更新、节点加入及退出都用到了多播通信。由于多播通信的不可靠性，发到一个组的消息可能不会到达所有的节点[15]。针对这一问题，现有文献提出了许多具有自愈能力的方案[9,10,15]。由于自愈能力的实质是提前几个安全会话(session)[9,10]分发组密钥，例如，文献[9]提前 l+2 个，文献[10]提前 l 个，其中，l 是估计的安全会话个数。因此，一旦某一成员被俘获，那么多播组将来的 l+2 个或 l 个会话的密钥都将被攻破。重新分发这些组密钥的代价是相当高的，而且也同样存在多播不可靠的问题。所以本文不采用自愈的方法。

假设节点 φ 因为某种原因(例如，暂时与网络断开)没有收到与密钥更新相关的多播消息，或者说，节点 φ 是多播组 ω 的成员，但它的组密钥是过期的。那么，当 φ 与网络重新连接后，会出现 2 种情况：① φ 收到组密钥更新消息；② φ 收到该多播组的数据消息。由 3.3 节和 3.4 节可知，φ 由于没有最新的密钥变更戳 $h(\xi)$而不能验证多播源，所以它无法更新组密钥。对于第 2 种情况，φ 通过解密操作发现自己的密钥是过期的。所以，φ 再次与网络连通后，只通过被动地接受消息无法获得组密钥。于是，φ 缓存数据消息(如果收到的是数据消息)，并构造单播消息：

$$\left\langle \begin{matrix} requireKey, \varphi, ID_\omega \mid \\ mac((requireKey, \varphi, ID_\omega)K_{\varphi\lambda}) \end{matrix} \right\rangle \tag{10}$$

其中，$(\cdot)_{K_{\varphi\lambda}}$ 表示用 $K_{\varphi\lambda}$ 做加密运算；$K_{\varphi\lambda}$ 是只有节点 φ 和组管理者 λ 构成的多播组的组密钥。由 3.2 节可知，$K_{\varphi\lambda}$ 也同样由 φ 和 λ 自己计算得到，并且只有 φ 和 λ 能计算得到。所以，λ 与 φ 能够相互认证。认证通过后，λ 确认 φ 是否属于多播组 ω。如果属于，将有 2 种情况：①组成员没有变化；②组成员发生了变化，但是，φ 仍然是成员，记新组为 ω'。则分别构造回复消息：

$$\left\langle \begin{matrix} newKey, ID_\omega, h(\xi) \quad \mid \\ mac(newKey, ID_\omega, h(\xi))_{K_{\varphi\lambda}} \end{matrix} \right\rangle \tag{11}$$

$$\left\langle \begin{matrix} newKey, h(\xi'), \ \omega' \quad \mid \\ mac(newKey, h(\xi'), \ \omega')_{K_{\varphi\lambda}} \end{matrix} \right\rangle \tag{12}$$

φ 收到后，同样可以认证该消息是否来自 λ，这样 φ 获得新密钥。在上面的过程中，λ 无论在哪次认证过程中没有通过，它都退出处理，不再回复。φ 在认证过程中没有通过，则等待，到时后，删除消息。

本方案通过单播单独联系的方式解决了多播不可靠问题。

5 安全性分析

5.1 组密钥生成方法的正确性

定理 1 由第 3 节方案生成的组密钥 K_ω 是正确的，或者说，设多播组 $\omega=\{\lambda, \varphi_1, \cdots, \varphi_\delta\}$，$\lambda$ 为发起者，则 $K_\omega = K_\lambda = K_{\varphi_2} = \cdots K_{\varphi_\delta}$ 成立。

证明 由对称多项式的性质易知，各节点利用式(5)计算的各 q 值从集合的角度看是相同的。又知二元运算符“$\wedge$”(“$\vee$”或“$\oplus$”)符合交换律。所以，各节点计算的 $h(\xi)\wedge q_{j_1}\wedge q_{j_2}\cdots\wedge q_{j_m}$ 值都是相等的，因此，由式(6)计算的各密钥值是相等的。

即有 $K_\omega = K_\lambda = K_{\varphi_2} = \cdots K_{\varphi_\delta}$ 成立。证毕。

5.2 组机密性

组机密性[10]是指非组成员无权访问组密钥。为了说明组密钥安全性，本节从多项式求解的角度分析在俘获 k+1 个节点的条件下，获得组密钥的难度。然后讨论变量 m 和 k 的取值。

首先讨论在一个多项式的情况下，即 m=1 时，俘获 k+1 个节点攻破多项式的过程。敌手要攻破一个多播组甚至整个网络的密钥体系，则它需要掌握分配给节点的多项式份额。由式(1)可知，即要获得 $a_{i_1,\cdots,i_t}$。便于说明问题，取 t=2，并令变量为 x 和 y，俘获的 k+1 个节点 ID 为 ID=1，…，k+1(这里俘获的节点不包括虚拟节点，只是为了说明方便，才将俘获节点的 ID 设置为 1，…，k+1)。展开式(1)得式(13)，其中，$a_{ij} = a_{ji}$。

$$\begin{aligned} & f(x,y) = \\ & a_{00} + a_{01}y + \cdots + a_{0k}y^k \\ & + a_{10}x + a_{11}xy + \cdots + a_{1k}xy^k \\ & \quad \vdots \\ & + a_{k0}x^k + a_{k1}x^k y + \cdots + a_{kk}x^k y^k \end{aligned} \tag{13}$$

则节点 ID=i 上预置的密钥份额为式(14)中系数 c_j^i 的集合 $C^i = \{c_0^i, c_1^i, \cdots, c_k^i\}$。反过来，从俘获的 k+1 个节点可知一个集合 $\{C^1, C^2, \cdots, C^{k+1}\}$，因此，可以构造 k+1 个具有 k+1 个变量的方程组。其中的一个如式(15)所示。这些方程组的变量是式(13)中的系数 a_{ij}，并且所有的系数矩阵是满秩的，所以可解。在预置一个多项式、俘获 k+1 个节点的情况下，按照上述过程，敌手能攻破密钥体系。

$$\begin{aligned} & g^i(y) = f(i,y) = \\ & a_{00} + a_{01}y + \cdots + a_{0k}y^k \\ & + a_{10}i + a_{11}iy + \cdots + a_{1k}iy^k \\ & \quad \vdots \\ & + a_{k0}i^k + a_{k1}i^k y + \cdots + a_{kk}i^k y^k \\ & = \sum_{j=0}^{k} c_j^i y^j \end{aligned} \tag{14}$$

$$\begin{cases} a_{00} + a_{10} \times 1 + \cdots + a_{k0} \times 1 = c_0^1 \\ a_{00} + a_{10} \times 2 + \cdots + a_{k0} \times 2^k = c_0^2 \\ \quad \vdots \\ a_{00} + a_{10}(k+1) \cdots a_{k0}(k+1)^k = c_0^{k+1} \end{cases} \tag{15}$$

但是，当节点预置多个对称多项式时，对于节点 i 来说，它具有多项式份额集合 $\{C_1^i, \cdots C_m^i\}$。由 3.1 节的步骤 4)可知，这些多项式份额是无序的。这样导致在构造式(15)时无法确切地知道俘获的 k+1 个节点上的哪些份额属于同一个多项式，出现了组合困难，见定理 2。

定理 2　记组合数为 com_m^k，则有式(16)。

$$com_m^k = (m!)^k \tag{16}$$

其中，k 为对称多项式的阶，m 为对称多项式的个数。

证明　这个问题的数学模型是点重新分组问题。即有 k+1 个组，每组含有 m 个点，所有的 $(k+1) \times m$ 个点都不同。从每一个组中取一个点组成一个新组，最终形成含有 k+1 个点的 m 个新组。这种重新分组的方法有多少？数学模型的元素与该问题的元素对应关系为：旧组代表被俘获的传感器节点，点代表节点上的多项式份额 C^i，新组代表属于同一个多项式的那些份额。

点重新分组问题解法：第 1 步，将第 1 个组的点(m 个)分到 m 个新组中；第 2 步，将第 2 个组的点做全排列 $m!$，每一种排列对应一种取法，与前一步合并后，每个组有 2 个点，有 $m!$种分组方法；第 3 步，任取一种前面 $m!$种分组方法，将第 3 个组的点做全排列 $m!$，进行组合，此时，每个组有 3 点，$m! \times m!$ 种分组方法。以此类推，直到最后一组。每组有 k+1 个点，有 $(m!)^k$ 种分组方法。

分析式(16)可知，组合困难对 m 和 k 的变化率很大，如表 1 所示，很小的 m 和 k 就会使组合数很大。正是由于这种组合困难，使获得节点的数量与攻破密钥体系之间的关系松散，从而使恶意节点的共谋攻击变得困难。OLA 根据式(16)和安全级别，确定合适的 m 和 k。

表 1　组合困难随 m 和 k 的变化情况

m	k=2	k=3	k=4	k=5
2	4	8	16	32
3	36	216	1296	7 776
4	576	13 824	331 776	8.0×10^8
5	14 400	1.7×1^0	2.0×10^8	2.4×10^{10}
6	518 400	3.7×10^8	2.7×10^{11}	2.0×10^{14}
7	2.5×10^7	1.3×10^{11}	6.5×10^{14}	3.2×10^{18}
8	1.6×10^9	6.6×10^{13}	2.6×10^{18}	1.1×10^{23}

5.3 前、后向安全性的保证

定义 3 前向安全性是指与多播组脱离的节点不能访问多播组后续组密钥的安全属性，以保证脱离的节点不能解密后续的组数据[10]。

定义 4 后向安全性是指新加入多播组的节点不能访问多播组先前组密钥的安全属性，以保证新节点不能解密以前的组数据[10]。

定理 3 节点加入或退出多播组会引起组密钥更新，3.4 节给出的组密钥更新方案能够保证前向、后向安全性。

证明 进一步分析 3.2 节组密钥建立过程中的式(5)和式(6)可知，该过程可以抽象为规则 3 和规则 4。规则 3 是计算过程的抽象。规则 4 是安全属性的抽象。该安全属性是指每个节点都有自己特定的多项式份额，从而使得组密钥只能由合法组成员计算得到，而非法组成员不可能得到。

规则 3 在过程上，组密钥是组成员 ID 和密钥变更戳 $h(\xi)$ 的函数，记为式(17)。

$$k = K(h(\xi),\varphi,\cdots) \tag{17}$$

规则 4 只有合法组成员才能得到正确的式(17)，非法组成员不能。

1) 前向安全性的保证：即证明脱离多播组的节点不能计算出新的组密钥。设原多播组由节点 $\omega=\{\lambda,\varphi_1,\cdots,\varphi_i,\cdots,\varphi_\delta\}$ 组成，假设 φ_i 退出多播组，则新的多播组由节点 $\omega'=\{\lambda,\varphi_1,\cdots,\varphi_{i-1},\varphi_{i+1},\cdots,\varphi_\delta\}$ 组成。由规则 3 得式(18)和式(19)。

$$K_\omega = K(h(\xi),\lambda,\varphi_1,\cdots\varphi_i,\cdots\varphi_\delta) \tag{18}$$

$$K_{\omega'} = K(h(\xi^{'}),\lambda,\varphi_1,\cdots\varphi_{i-1},\varphi_{i+1}\cdots\varphi_\delta) \tag{19}$$

因为 $K_{\omega'}\neq K_\omega$，而且脱离的节点 φ_i 即使获得了新的密钥变更戳 $h(\xi^{'})$，由于自己是非法的组成员，由规则 4 可知，φ_i 也不能获得正确的式(19)，从而无法计算得到新的组密钥 $K_{\omega'}$，保证了前向安全性。

2) 后向安全性的保证：同理，设原多播组由节点 $\omega=\{\lambda,\varphi_1,\cdots,\varphi_i,\cdots,\varphi_\delta\}$ 组成，新加入的节点为 φ_i，则新的多播组由节点 $\omega=\{\lambda,\varphi_1,\cdots,\varphi_i,\cdots,\varphi_\delta\}$ 组成。由规则 3 得式(20)和式(21)。

$$K_\omega = K(h(\xi),\lambda,\varphi_1,\cdots\varphi_\delta) \tag{20}$$

$$K_{\omega'} = K(h(\xi^{'}),\lambda,\varphi_1,\cdots\varphi_i,\cdots\varphi_\delta) \tag{21}$$

因为 $K_\omega\neq K_{\omega'}$，而且新节点 φ_i 即使获得旧的组密钥变更戳 $h(\xi)$，由规则 4 可知，新节点 φ_i 也无法计算旧的组密钥 K_ω，保证了后向安全性。证毕。

由于组密钥更新秘密(ξ)或者密钥变更戳($h(\xi)$)的引入，使得即使同一组节点再次组成一个多播组，也会有不同的组密钥。认为由相同节点组成的多播组就是同一个多播组，具有相同的组标识。

5.4 密钥更新过程安全性分析

3.3 节和 3.4 节过程相似，都涉及到密钥更新问题，分析如下。

由组密钥管理模型可知，只有多播组管理者(或多播组的发起者)才有权限更新密钥。因此，组成员在更新自身的组密钥时，要对更新消息进行认证，防止伪造的、重放的组密钥更新消息，即进行多播源认证。下面分析多播源认证的安全性。

成功变更密钥的关键是组密钥更新消息中是否包含正确的密钥更新秘密 ξ。由 3.2 节组密钥建立过程可知，多播组管理者在发起多播组时，加入了密钥变更戳 $h(\xi)$。由 $h(\cdot)$的单向性可知，无法通过 $h(\xi)$ 求得 ξ。因此，除了组管理者之外，没有任何节点知道组密钥更新秘密 ξ。从而也就无法伪造组密钥更新消息(8)和(9)，于是挫败了伪造攻击。又因为每次更新密钥时，都会产生新的密钥更新秘密 ξ' 和密钥变更戳 $h(\xi')$，因此，密钥变更戳 $h(\xi')$ 具有临时值的作用，隐式地保证了密钥更新消息的新鲜性，挫败了重放攻击。

5.5 俘获组管理节点对多播组的影响

在异构模型中，多播组管理节点是基站或密钥中心等中心控制结构，实际上会导致单点失败，但是，相关的方案都无一例外地假定了该中心点是安全的，回避了该问题。无论是异构模型还是同构模型下，俘获多播组管理节点对于多播组的安全性都是致命的，因为管理者掌握着密钥建立、更新等关键过程。但是，同构模型下的情况好得多。原因如下：同构模型下的多播组是以多播组管理节点的业务为中心的，管理节点被俘获，业务就消失，组的安全机制也就失去了意义。管理节点被俘获对于业务的打击比对安全的打击严重得多，因此，对多播组安全机制的影响可以不考虑。重要的是，该多播组的其他节点仍就可以服务于其他的多播组，其他的多播组依然可以安全工作。即对一个组的影响不会波及到其他的组，不会导致全部多播组的瘫痪。而对异构网络下的多播组却是致命的，原因是所有的多播组都依靠同一个中心节点，这一中心点的失败必然会导致所有依赖的多播组瘫痪。这是分布式

动态多播组比中心式(或异构网络)多播组具有优势的一个方面。

6　性能分析与比较

本节定量分析方案的性能，并与文献[12](记为 D 方案)进行比较。

6.1　组密钥管理模型实现程度比较

由 3.2 节的组密钥建立过程可知，由于共享了对称多项式，所以任意大于 2 的任意节点可以计算组密钥，而不增加存储空间和无线通信消耗。对节点数量和节点身份没有任何限制，因而是动态的。

本方案区分了不同的多播组，每个组都有自己的标识，从理论上说，节点参加多播组或发起多播组只取决于网络需求。从操作角度看，节点参加一个组或发起一个组所需要的存储空间、计算复杂度和通信消耗十分有限，因此，也是可行的。由于方案没有利用任何网络拓扑信息或节点位置信息，并且任意数量的任意节点可以协商密钥，所以，方案支持节点移动。处理多播通信不可靠性问题，请参照 4.4 节。可扩展性的论述，请参照第 6.2 节。综上所述，方案满足 2.2 节提出的动态组密钥管理模型的所有要求。表 2 显示了 D 方案和本方案对组密钥管理模型实现情况的比较。

表 2　　2 种方案实现程度的比较

模型要求	D 方案	本方案
动态性	×	√
多角色	×	√
成员关系可变更性	√	√
移动性	×	√
健壮性	/	√
可扩展性	√	√

说明：“×”表示没有实现，“√”表示实现，“/”没有这样的指标。“模型要求”中的模型是指 2.2 节中提出的组密钥管理模型。由于 D 方案在进行密钥管理时全部采用单播方式，所以“健壮性”为“/”。

6.2　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。

D 方案将整个网络看作一个组，目标是使网络共享一个密钥。为了支持节点追加的能力，父代节点需要预置与各子代节点进行通信的密钥材料。同代节点需要预置 2 个密钥：组认证密钥(group authentication key) bk_i 和密钥产生密钥(key generation key) bk_i。不同代节点间建立安全连接需要预置 2 个密钥和一个随机数。分别为：$S_{i,x}$、$gk_{i,x}$ 和 R_i、上述的 i, x 表示部署序号。

下面分析 2 种方案的存储使用情况。分析 D 方案可知，各代节点需要预置的密钥数与代数有关。设共有 π 代节点，即部署 1 次，追加 $\pi-1$ 次。则 D 方案各代节点需要预置的密钥数为

$$(\pi+2)l \tag{22}$$

由第 3 节可知，本方案在部署前要求节点预置的数据为

$$m(k+1)l \tag{23}$$

其中，m 为多项式的个数，最小值为 2，k 为多项式的阶。比较式(22)和式(23)可知，本方案预置数据的数量与节点的代数没有关系，对节点追加的次数没有限制。而 D 方案预置数据的数量与节点追加的次（代）数有线性关系。因此，网络的可扩展性受到限制（如图 1 所示）。

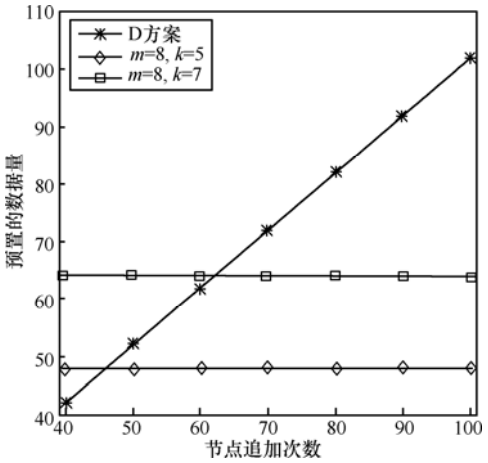

图 1　预置数据量与节点追加次数的关系

6.3　计算量、通信量分析

由于每种传感器做相同的计算或者产生相同无线通信量所用的能耗并不同，所以本文在进行能耗分析时，以计算量和通信量为单位。计算量用运算类型衡量，通信量用消息数量和大小衡量。

表 3 显示了 D 方案与本方案计算量和通信量的比较结果。表中的数据是进行组密钥操作时网络的整体能量消耗。实验[16]显示发送和接收相等的数据量能耗是不同的，所以在表 3 中分别考虑了发送与接收的数据量。

表 3 中各参数的意义如下。

h 表示散列运算；mac 表示消息认证码运算；

ploy 表示多项式运算；*l* 表示节点 ID、多项式系数、密钥、随机数等的长度；*n* 表示网络规模；*δ*表示动态组的节点个数；*δ'*表示新加入的或强制退出的节点个数；TX 表示发送，RX 表示接收，后跟数据量；“/”表示没有这项内容。

表 3　　计算量、通信量分析及比较

组密钥操作	代价类型	D 方案	本方案	
			λ	φ
建立	计算	$2mac \times n$	$2h+mac$	$(mac + h + m \cdot poly) \times \delta$
	通信	TX: $(6l+1) \times n$; RX: $(6l+1) \times n$	TX: $(3+\delta)l+1$	RX: $((3+\delta)l+1) \times \delta$
更新	计算	$2mac \times n$	$h+mac$	$(mac + 2h + m \cdot poly) \times \delta$
	通信	TX: $(6l+1) \times n$; RX: $(6l+1) \times n$	TX: $4l+1$	RX: $(4l+1) \times \delta$
节点加入	计算	$8mac \times \delta'$	$h+mac$	$(mac + 2h + m \cdot poly) \times (\delta + \delta')$
	通信	TX: $(10l+2) \times \delta'$; RX: $(12l+3) \times \delta'$	TX: $(3+\delta+\delta')l+1$	RX: $((3+\delta+\delta')l+1) \times (\delta+\delta')$
节点退出	计算	$2mac \times n$	$h+mac$	$(mac + 2h + m \cdot poly) \times (\delta - \delta')$
	通信	TX: $((6+\delta')l+1) \times n$; RX: $((6+\delta')l+1) \times n$	TX: $(3+\delta-\delta')l+1$	RX: $((3+\delta-\delta')l+1) \times (\delta-\delta')$
多播可靠性维护	计算	/	$2mac$	$2mac+2h+m \cdot poly$
	通信(单播)	/	$3l+1$ / $(2+\delta-\delta')l+1$	$3l+1$

表 3 表明，在进行组密钥相关操作时，本方案比 D 方案使用更少的网络能耗。原因是 D 方案每种组密钥操作都与全体节点有关，而本方案只把操作限制在组成员范围内，因而是高效的。

7　结束语

本文针对无线传感器同构网络中动态形成的多播组给出了完整的组密钥管理模型，并提出一种满足该模型的组密钥管理方案。该方案不需要中心式控制设施(如基站或密钥服务器 KS)，依靠动态多播组成员相互协作完成密钥管理功能，因而与现有方案是不同的。与同类方案相比(如文献[11]和文献[12])，本方案有如下特点：①支持的多播组的大小及数量是可变的；②节点是可移动的；③网络是可扩展的；④大大增加了大规模节点成功共谋的难度。因此本方案更加灵活、完备、安全。性能分析表明，本方案比 D 方案具有更好的存储、计算和通信性能。所以，本文提出的组密钥管理方案更适用于资源受限的同构无线传感器网络。

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, 30:2314-2341.

[2] 袁斑,马建庆,钟亦平等.基于时间部署的无线传感器网络密钥管理方案[J]. 软件学报, 2010, 21(3):516-527.
YUAN T, MA J Q, ZHONG Y P, *et al*. Key management scheme using time-based deployment for wireless sensor networks[J]. Journal of Software, 2010, 21(3):516-527.

[3] 苏忠,林闯,封富君等.无线传感器网络密钥管理的方案和协议[J]. 软件学报,2007,18 (5): 1219-1231.
SU Z, LIN C, FENG F J, *et al*. Key management schemes and protocols for wireless sensor networks[J]. Journal of Software, 2007, 18 (5):1219-1231.

[4] JR M S, BARRETO P, MARGI C B, *et al*. A survey on key management mechanisms for distributed wireless sensor networks[J]. Computer Networks, 2010, 54: 2591-2612.

[5] WANG E K, YE Y. An efficient and secure key establishment scheme for wireless sensor network[A]. Third International Symposium on Intelligent Information Technology and Security Informatics[C]. 2010.511-516.

[6] 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, 2003.

[7] 黄海平,王汝传,孙力娟等.基于密钥联系表的无线传感器网络密钥

管理方案[J].通信学报，2006，27(10):13-18.
HUANG H P, WANG R C, SUN L J, *et al*. New key management scheme of wireless sensor networks based on key relation table [J]. Journal on Communications, 2006, 27(10): 13-18.

[8] 李凤华，王巍，马建峰. 适用于传感器网络的分级群组密钥管理[J]. 电子学报, 2008, 36(12): 2405-2411.
LI F H, WANG W, MA J F. Leveled group key management for wireless sensor networks[J]. Acta Electronica Sinica, 2008, 36(12): 2405-2411.

[9] 李林春,李建华，潘军.无线传感器网络中具有撤销功能的自愈组密钥管理方案[J]. 通信学报, 2009, 30(12):13-17.
LI L C, LI J H, PAN J. Self-healing group key management scheme with revocation capability for wireless sensor networks[J]. Journal on Communications, 2009, 30(12):13-17.

[10] SHI M H, SHEN X M, JIANG Y X, *et al*. Self-healing group-wise key distribution schemes with time-limited node revocation for wireless sensor networks [J]. Ad hoc Networks, 2007, 5(1):14-23.

[11] BLUNDO C, SANTIS A D, HERZBERG A, *et al*. Perfectly-secure key distribution for dynamic conferences[J]. Lecture Note in Computer Science, 1993, 740:471-486.

[12] DUTERTRE B, CHEUNG S, LEVY J. Lightweight key management in wireless sensor networks by leveraging initial trust[EB/OL]. http://en.scientificcommons.org/56845498, 2004.

[13] 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, 2010.1-5.

[14] WANG Y, WANG X D, XIE B *et al*. Intrusion detection in homogeneous and heterogeneous wireless sensor networks[J]. IEEE Transactions on Mobile Computing, 2008, 7(6):698-711.

[15] LIU D G, NING P, SUN K. Efficient self-healing group key distribution with revocation capability[A]. Proceedings of the 10th ACM Conference on Computer and communications security[C]. Washington, DC, USA, 2003.231-240.

[16] ZHANG Y, LIU W, LOU W, *et al*. Location based security mechanism in wireless sensor networks[J]. IEEE JSAC, 2006, 24(2):247-260.

作者简介：

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。

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。

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。