第 33 卷第 12 期 通 信 学 报 Vol.33 No. 12
2012 年 12 月 Journal on Communications December 2012
doi:10.3969/j.issn.1000-436x.2012.12.008

两层传感器网络中一种高效的加密数据条件聚合协议研究

李睿，林亚平，李晋国
(湖南大学 信息科学与工程学院，湖南 长沙 410082)

摘 要：提出了一种隐私保护的条件聚合协议，使存储节点在不知道数据真实值的情况下对满足条件的数据进行聚合，防止存储节点对敏感信息的泄漏。为了保护数据和查询条件的隐私性，提出了一种基于前缀成员确认和布鲁姆过滤器相结合的编码方法对数据和查询条件进行编码，实现存储节点在不知道数据真实值和查询条件真实值的情况下进行查询处理；为了对查询结果中的数据进行聚合而不暴露数据真实值，采用同态加密技术对数据进行加密，使数据在不解密的情况下能进行聚合运算。进一步，根据传感器采集数据的特点，提出了一种基于代码表的数据压缩表示及传输方法，有效减小了传感器节点和存储节点之间的通信开销。分析和实验结果验证了所提方案的有效性。

关键词：两层结构传感器网络；安全范围查询；加密数据聚合；条件聚合；代码表
中图分类号：TP393 **文献标识码：**A **文章编号：**1000-436X(2012)12-0058-11

Efficient conditional aggregation of encrypted data in tiered sensor networks

LI Rui, LIN Ya-ping, LI Jin-guo
(College of Information Science and Engineering, Hunan University, Changsha 410082, China)

Abstract: A privacy preserving conditional aggregation protocol was proposed that enabled storage nodes to aggregate the data items satisfied sink issued queries correctly while prevented them from revealing both sensor collected data and Sink issued queries. To protect privacy for sensor col data and sink issued queries, an encoding method based on prefix membership verification and bloom filters was proposed to encode both sensor collected data and sink issued queries, which allowed storage nodes process queries correctly without knowing their actually value. Homomorphism encryption method was adopted to encrypt sensor collected data, which enabled storage nodes to process aggregation on encrypted data items. To reduce corresponding energy consumption between sensors and storage nodes, a code table method was proposed to represent and transmit data items. Analysis and experime validate the efficacy and efficiency of the proposed protocol.

Key words: two-tiered sensor networks; secure range queries; encrypted data aggregation; conditional aggregation; code tables

1 引言

传感器网络为环境和移动目标的监测及数据处理提供了一种非常经济的分布式解决方案[1,2]，两层结构传感器网络(two-tired sensor networks)已被认为是无线传感器网络未来的发展趋势[3~6]。图 1

收稿日期：2012-07-10；修回日期：2012-10-18
基金项目：湖南省自然科学基金资助项目(12JJ3068)；湖南省科技计划基金资助项目(2012TT2054)；国家自然科学基金资助项目(61173038)；湖南大学“青年教师成长计划”基金资助项目
Foundation Items: The Natural Science Foundation of Hunan Provience (12JJ3068); Hunan Provincial Scienc & Technology Plan Project (2012TT2054); The National Natural Sciece Foundation of China (61173038); Young Teachers Growth Plans of Hunan University

所示是一个典型的两层传感器网络。在该网络中包含大量的资源受限的传感器节点以及数量相对较少的存储节点。传感器节点处在网络中的下层，负责收集监测目标数据，并周期性地将收集到的数据提交给附近的存储节点。存储节点处在网络中的上层，为传感器节点提供数据存储服务的同时也负责处理 sink 节点发来的查询。

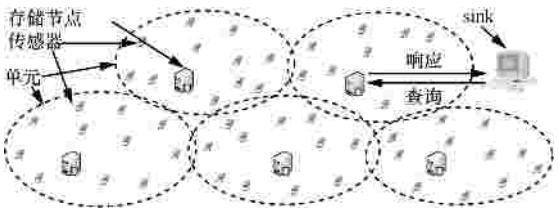

图 1　两层结构的传感器网络的体系结构

在网络中，传感器节点收集了大量监测目标的细节数据，而用户查询结果往往需要对满足查询条件的细节数据进行综合计算后得出。例如，用户可能对一个用于环境监测的传感器网络提出如下查询请求：

select average humidity where time ∈ [11：00pm- 2：00 am] and temperature ∈[20℃~25℃] and sense area is A_1。

该查询要求计算满足时间、温度和感知区域条件下湿度的平均值。如果将满足条件的所有细节数据全部发送到 sink 节点后再进行计算，则在传感器网内会产生大量的数据流。为了减小网内数据流，降低节点的能耗，研究者提出了数据聚合（data aggregation）[7,8]的概念。数据聚合运算通过网内节点对来自不同数据源的数据进行提前综合，以此来去除数据冗余，减小数据发送量，达到节能和延长网络寿命的目的。由于传感器节点计算能力的受限，传统传感器网络中的数据聚合往往只能针对一个周期内传感器节点收集的所有数据进行，而没有考虑对跨周期数据以及通过条件筛选后的数据进行聚合。在两层传感器网络中，存储节点收集了附近传感器节点的多个周期内数据，并且具有相对丰富的计算资源，为复杂聚合运算提供了条件。为区别传统传感器网络下的数据聚合运算，本文称对满足查询条件下数据的某一个属性值进行聚合的运算为条件聚合运算。在两层传感器网络中，存储节点虽然比传感器节点具有更丰富的能量，但它们需要承担更多的计算任务，并且一个存储节点的失效会导致所在传感器区域成为检测盲区。因此，研究两层传感器网络中的条件聚合对延长网络的生命周期具有非常重要的意义。

然而，存储节点为条件聚合提供了条件的同时也带来了新的挑战。由于其在网络中扮演的关键角色导致在敌对环境下更容易招致攻击。一个妥协的存储节点会泄漏存储在该存储节点上的敏感数据。如何在两层传感器网络上设计一种安全条件聚合协议，使存储节点在不泄漏传感器采集的敏感数据的情况下进行聚合是一项挑战性的任务。其挑战性表现在以下几个方面：第一，为了防止敏感数据的泄漏，需要对传感器节点采集的数据和 sink 发送来的查询条件进行加密处理，因此，需要解决在不知道数据真实值和查询条件真实值的情况下实现查询处理；第二，要解决对满足查询条件的加密数据在不解密的情况下实现聚合运算；第三，为了对传感器采集数据的某一个属性进行聚合，需要对属性值单独进行加密，消耗更多空间，因此要解决传感器节点和存储节点之间的数据通信开销问题。

安全条件聚合包含安全查询和安全聚合两方面工作。安全查询获得满足条件的细节数据；安全聚合对满足条件的细节数据进行聚合。目前还没有相关工作研究两层传感器网络当中的条件聚合查询。与本文最相关的工作是两层传感器网络中的安全范围查询[9~12]和传统传感器网络中的数据聚合[13~15]。

在两层传感器网络中的安全查询方面，最具代表性的工作是 Chen&Liu 提出的 SafeQ[12]安全查询协议。SafeQ 考虑的安全模型是传感器和 sink 是安全的，攻击者只妥协存储节点。在他们设定的模型下，SafeQ 具有优良的性能，特别是在多维数据的情形。但如果攻击者妥协了一个存储节点的同时再妥协该存储节点所在单元的一个传感器节点，则所有存储在该妥协存储节点上的数据都会泄露。其原因如下：在 SafeQ 中，同一个单元内的所有传感器共享一个密钥用于数据的编码，并且每一个传感器采集的数据按照有序的方式组织。因此，攻击者妥协一个传感器后可以获得共享密钥，并控制妥协的传感器按照需要产生数据，通过二分查找法可以查找出所有其他传感器采集的数据值。文献[9~11]是基于桶的隐私和完整性认证方法。桶的方法最主要缺陷有两个：①攻击者可以较容易估计出传感器检测区域数据的分布情况，这样部分暴露了敏感信息；②当传感器

采集的数据集中在少数的几个桶中时，传感器和存储节点都会消耗过多的能量和空间。同时，基于桶的方法不适合本文的条件聚合查询，因为往往一个桶中只有部分数据满足查询条件。

在传统无线传感器网络中的安全聚合方面，最具代表性的工作是 Przydatek 等人提出的 SIA（secure information aggregation）聚合机制[13]。在 SIA 机制中，传感器节点通过共享密钥对数据进行加密后传给汇聚节点，汇聚节点用共享密钥解密后进行聚合，并将聚合后的结果重新进行加密传给上层汇聚节点。SIA 能有效防止外部攻击者的窃听，但无法阻止内部妥协的汇聚节点对数据隐私性的泄露。针对这一问题，Castelluccia 等人采用同态加密技术实现对密文数据的聚合[15]。由于传感器节点本身资源受限，传统的无线传感网络中的聚合只考虑将传感器节点在一个周期内所采集的所有数据进行聚合，没有考虑跨周期数据的聚合以及对满足条件的数据进行聚合。

为此，本文提出了一种基于两层传感器网络的安全条件聚合运算协议。该协议包含两部分：在不泄漏数据隐私的情况下实现范围查询以及对符合查询条件的数据进行聚合。为了在保护数据隐私的情况下实现查询，本文采用前缀成员确认技术和布鲁姆过滤器相结合的办法对数据进行编码，实现了存储节点在不知道数据真实值和查询条件真实值的情况下进行查询处理；为了在保护数据隐私的情况下实现数据聚合，采用同态加密技术，使数据在不解密的情况下可以进行相关的聚合运算。另外，针对经过编码和加密后的数据所占空间较大的问题，在分析传感器采集数据更新率的基础上提出了一种基于代码表的数据压缩表示与传输方法。该方法将传感器已经计算过的编码和加密结果保存在代码表中，并且将该代码表同步到存储节点上，当重复数据出现时，只需要传输相应的代码，存储节点根据代码在本地代码表中检索出相应的编码和加密结果。分析和实验结果证实了该方法能有效减少数据通信量以及避免传感器节点的重复计算。

本文的主要贡献如下：①在两层传感器网络当中提出了隐私保护的条件聚合问题；②提出了一种基于前缀编码和布鲁姆过滤器相结合的数据隐私编码方案；③提出了一种基于代码表的数据压缩表示和传输方法。

2 模型

2.1 系统模型

如图 1 所示，一个典型的两层结构传感器网络包括 3 种类型的节点：传感器、存储节点和 sink。传感器节点是一些资源受限的节点，在网络中采集数据并周期性地将采集到的数据提交给附近的存储节点。存储节点相比传感器节点具有更丰富的资源，在网络中存储节点存储传感器发送过来的数据，并根据存储在其上的数据处理 sink 节点发送过来的查询。Sink 节点是用户访问传感器网络的接入点，负责处理用户提出的问题。当接收到用户的一个问题后，Sink 首先将该问题改写成符合网络特点的查询，并将这些查询发送给与该问题相关的存储节点。收到存储节点发回的查询结果后，Sink 将查询结果进行综合得到最终的问题答案发送给用户。

2.2 基本假设

对于上述两层传感器网络，本文做出如下假设。

1) 传感区域分成很多个单元(cell)，一个单元中有多个传感器节点和一个存储节点。每一个传感器节点和存储节点知道节点本身的位置及其所在的单元。所有的传感器在部署前分配一个唯一的 ID。

2) 所有传感器与存储节点均是松散同步，本文将时间划分成互不重叠的周期，每个周期内节点采样的次数为 n 次，传感器 s_h 所采集的数据用三元组 $(h,t,\{d_1,d_2,\mathrm{L},d_n\})$ 表示，其中，t 表示周期序号，d_1,d_2,L,d_n 表示传感器采集的数据；传感器在周期末向所在单元中的存储节点发送数据。

3) 所有传感器与 sink 共享一个相同单向散列函数 $H(\cdot)$；每一个传感器节点预置一个不同的编码布鲁姆过滤器（包括散列表长度，散列函数个数以及散列函数本身）；每一个传感器节点和 sink 共享一个密钥，如传感器 s_h 与 sink 共享的密钥为 k_h。

2.3 威胁模型

本文研究如何防止攻击者从传感器网络中获得敏感数据信息而破坏数据的隐私性。在很多实际应用中，保护敏感数据隐私性是一个关键任务，如在用于监测健康状况的传感器网络中，个人健康数据的保护就显得非常重要。

假设 sink 是可信的，存储节点可能被妥协，少量传感器节点也可能被妥协。另外，本文与文献[16]一样采用半可信[17]威胁模型，即妥协的存储节点在数据处理过程依然严格遵守协议，但试图破坏数据

的隐私性。本文不考虑妥协的传感器节点伪造自己的数据。主要有两个原因：①很难阻止妥协的传感器伪造自己的数据；②当限制每一个传感器在一个周期内发送的数据量时，少量传感器妥协对最终查询结果的影响较小，文献[18，19]指出，如果传感器节点只伪造自己的数据，查询结果离真实结果的偏差不会太大。

3　数据加密方案

文献[16]对在两层传感器网络当中的最大、最小值的聚合问题进行了探讨。本文研究平均值、方差以及求和等聚合运算在两层传感器中的实现。为了保护数据的隐私性，存储节点需要在密文的情况下对数据进行聚合。同态加密中对密文的某种特定的线性运算与对明文另一种特定的线性运算是等价的，它可以用来实现密文的聚合运算，因此本文采用同态加密技术对需要进行聚合的数据的属性值进行加密。

同态加密的思想如图 2 所示。用 $C=Enc(K，V)$ 表示用密钥 K 对明文 V 加密得到密文 C，令 $C_1=Enc(K_1，V_1)$，$C_2=Enc(K_2，V_2)$，根据同态性质有 $C_1+C_2=Enc(K_1+K_2，V_1+V_2)$。根据这一性质，第三方可以对密文数据进行运算得到所需要的运算结果。

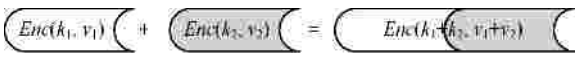

图 2　同态加密运算的基本思想

由于无线传感网络中传感器节点资源受限，因此进行同态加密时运算不能太复杂。文献[20]给出了一种轻量级加法同态加密算法，该算法基于模运算对加法的可分配性性质，即：当 $a+b<M$ 时，有 $(a \bmod M)+(b \bmod M)=(a+b) \bmod M$。该同态加密算法能满足本文的需求。下面简述该同态加密算法的基本原理以及密文下的求和、均值和方差等聚合运算的具体实现以及相应的解密过程。

该同态加密运算的加、解密基本思想为： 选取一个大整数 M 以及密码 $K(K<M)$，对明文 p 计算 $C=(p+K) \bmod M$ 作为其加密结果。由于密钥 K 以及 M 的秘密性，攻击者在只已知密文的情况下无法破解出明文。要进行解密时，计算 $(C- K) \bmod M$ 得到相应的明文。例如，密钥 $K=28$，明文 $p=7$，大整数 $M=30$，当要加密 p 时，计算 $C=(p+K) \bmod M= (7+28) \bmod 30=5$。要解密 C 时，计算 $(C- K) \bmod M =(5- 28) \bmod 30=7$，得到明文。

由于模运算满足对加法的分配律，因此该加密算法满足加法同态。假设对明文 p_1 和 p_2 进行加密的密钥分别为 K_1 和 K_2，其加密数据的和与求和后进行加密的结果在模运算下一致，即满足：$C=(C_1+C_2) \bmod M= (\mathrm{Enc}(K_1, p_1)+ \mathrm{Enc}(K_2, p_2)) \bmod M = \mathrm{Enc}(K_1+K_2, p_1+p_2) \bmod M$（注：第三方进行加密数据的运算时，只需要计算 $\mathrm{Enc}(K_1, p_1)+ \mathrm{Enc}(K_2, p_2)$的结果，解密方已知 M 和密钥可以正确进行解密运算）。当 $p_1+p_2<M$ 时，有 $(p_1 \bmod M)+(p_2 \bmod M))\bmod M=(p_1+p_2) \bmod M$，因此要正确解密出 p_1+p_2，要求选取的 M 比参加运算的明文的总和大。当参加同态加密运算的数据个数为 n，其中最大的元素为 $V_{\max}$，一般要求 $M>nV_{\max}$。

该同态加密不支持除法运算，因此计算数据的平均值时，需要将数据进行解密后除以参加运算的数据个数得出。接下来讨论方差的计算。方差的计算公式为

$$s^2=\frac{(x_1-\bar{x})^2+(x_2-\bar{x})^2+\cdots+(x_n-\bar{x})^2}{n}=\frac{(x_1^2+x_2^2+\cdots+x_n^2)}{n}-\bar{x}^2$$

因此，为了计算方差，进行同态加密时，需要计算出明文的平方值，然后分别对明文和明文的平方值采用同态加密算法进行加密。当参加同态加密运算的数据个数为 n，其中，最大的元素为 $V_{\max}$，进一步要求 $M>nV_{\max}^2$。n 为可支持的参加同态计算的密文个数，该参数将在分析部分和实验部分进行讨论和实验。当要求参加聚合运算的加密数据个数超过 n 时，将参加运算的数据分划分成最少的子集，使得每一子集中所含元素个数不超过 n，然后分别进行计算，将结果发送给解密方，解密方进行解密后获得这些子集的解，再进行后续运算。

在两层传感器网络中，传感器节点和 sink 共享加密密钥。传感器将数据进行加密后发送给存储节点，存储节点对满足条件的所有传感器节点发送的数据进行聚合运算，并将运算的结果发送给 sink。Sink 根据参加聚合运算的数据个数以及数据提供者的 ID 进行解密得到相应的明文。因此，数据加密方是传感器节点，数据解密方是 sink 节点，密文的运算在存储节点上进行。

4 数据隐私查询编码方案

数据的条件聚合是针对满足条件的数据进行聚合，采用同态加密可以在不解密数据的情况下实现数据的聚合运算，如何在不知道数据真实值的情况下判断数据是否满足相应的查询条件是一个难点。本文中考虑了一个更为复杂的情况，也就是攻击者妥协存储节点后，可能还会妥协一个或者几个传感器节点，妥协后的传感器节点不能威胁其他传感器节点采集的数据。

本文数据编码的基本思想是：假设传感器 s_h 在周期 t 内采集的数据是 $\{d_1,d_2,\cdots,d_n\}$。s_h 对数据采用与 sink 共享的密钥 k_h 进行加密后，应用 2 个函数 $H(\cdot)$ 和 $X_h(\cdot)$ 对数据进行编码，如数据 d_j 的编码结果为：$X_h(H(d_j))$。其中，$H(\cdot)$ 是一个单向函数，为所有传感器节点共享；$X_h(\cdot)$ 是 s_h 独享的一个秘密函数。在一个周期结束时，s_h 将加密和编码后的数据一起发送给存储节点。当需要进行一个范围查询 $\{t,[a,b]\}$ 时，Sink 应用另外一个函数 $M(\cdot)$ 对查询范围进行处理，得到 $M([a,b])$。为防止妥协的传感器节点伪造查询条件，sink 对查询条件 $\{t,M([a,b])\}$ 进行数字签名后发送给相应的存储节点。当存储节点需要对传感器节点 s_h 提交的数据进行查询时，存储节点将 sink 发来的查询条件 $\{t,M([a,b])\}$ 发送给传感器节点 s_h。传感器节点 s_h 确认查询条件由 sink 产生后，采用秘密函数 $X_h(\cdot)$ 将该查询条件转换成 $\{t,X_h(M([a,b]))\}$ 并发送给存储节点。存储节点利用其上面的查询函数 $Q(\cdot)$ 实现对加密的数据进行查询。这些函数要满足以下条件：①数据 d_j 在查询范围 $[a,b]$ 中，当且仅当 $Q(X_h(B(d_j))X_h(M([a,b])))$ 为真。该条件保证存储节点在不知道数据真实值和查询条件真实值的情况下确定加密数据是否在查询结果中。②给定 $X_h(H(d_j))$ 和 $(d_j)_{k_h}$，存储节点在知道妥协传感器节点的信息后依然无法计算出数据 d_j；同时给定 $\{t,M([a,b])\}$ 和 $\{t,X_h(M([a,b]))\}$，存储节点也无法破解 $X_h(\cdot)$。该条件是保证传感器节点所采集数据的隐私。③给定 $M([a,b])$，存储节点和妥协的传感器节点都不能计算出查询条件 $[a,b]$。该条件保证查询条件的隐私性。图 3 给出编码方案的基本思想。

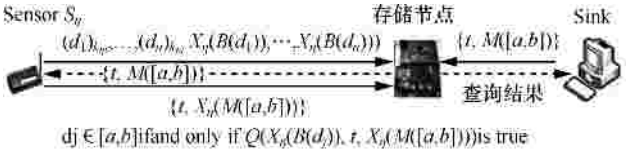

图 3 编码方案的基本思想

4.1 前缀成员确认

为了实现在存储节点妥协以及少量传感器节点妥协情况下能保护数据的隐私性，本文首先将范围查询转换成成员确定。成员确定采用前缀成员技术，该方法在文献[21]提出，在文献[22]中成形。前缀成员确认技术的关键思想是将判断一个数据是否在指定的范围中转换成判断一个集合和另外一个集合是否有交集的问题。以下叙述中数字的表示均采用二进制。用 $\{0,1\}^k\{*\}^{w-k}$ 表示前面具有 k 个先导字符“0”或“1”，然后紧跟 $w-k$ 个“*”的前缀，并称该前缀为 k 长度前缀。例如，“1****”是 1 长度前缀，该前缀表示的范围为[1000，1111]。当数据 x 在一个 k 长度前缀表示的范围中，称 x 与该前缀匹配。根据前缀的特点，数据 x 与一个 k 长度前缀匹配，当且仅当数据 x 的前 k 位与该 k 长度前缀的前 k 位完全一致。例如，x 与 1***匹配，则 x 的第一位必需是 1。如果前缀 P 表示的范围是前缀 Q 表示的范围的子集，称前缀 Q 是前缀 P 的祖先前缀。如前缀 1***是前缀 101*的祖先前缀。如果前缀 Q 是前缀 P 的最小祖先前缀，称前缀 Q 是前缀 P 的父前缀。如前缀 1***是前缀 10**的父前缀。称包含一个数的所有前缀的集合为该数的前缀科(prefix family)。给定一个数据 N，假设该数据采用二进制表示为：$b_1b_2\cdots b_w$。则该数据的前缀科是一个包含 $w+1$ 个前缀的集合：$\{b_1b_2\cdots b_w, b_1b_2\cdots b_{w-1}*,\cdots,b_1*\cdots*, **\cdots*\}$。用 $F(x)$ 表示数 x 的前缀科。产生一个数据的所有前缀是比较多的。如一个 32 位数的前缀科中含 33 个前缀，在很多情况下不需要产生所有的前缀。如需要处理的数据在给定的一个前缀表示的范围内，则所有该前缀的祖先前缀就显得不重要，因为该前缀表示范围内的所有数据共享这些祖先前缀。为了减小前缀的数量，定义前缀 R 下的数据 x 的前缀科(prefix family for x under prefix R)。对于数据 x，设该数据的前缀科为 $F(x)$，$F(x)$ 中所有不是前缀 R 的祖先前缀的前缀组成的集合称为数据 x 在前缀 R 下的前缀科，记作：$F_R(x)$。

前缀成员确认正是基于以下事实：给定前缀 R，任何数据 x 和前缀 P（R 是 P 的祖先前缀），x 在 P

中，当且仅当 P 在 $F_R(x)$中。为确认一个数据 x 是否在范围[a，b] ([a，b]是前缀 R 所确定的范围的子集)中，首先将范围转换成一个最小前缀集合，记作 $S([a，b])$，使得 $S([a，b])$中所有前缀表示的范围的“并”等于范围[a，b]。例如，$S([10，15])$={101*，11**}。计算数据 x 在前缀 R 下的前缀科 $F_R(x)$。x 在范围[a，b]中，当且仅当 $F_R(x)$n $S([a，b])?F$。

接下来讨论如何将前缀转换成数字方便上述集合的运算。本文采用文献[23]的方法将前缀分别转换成唯一对应的一个数字。具体方法为：采用数字“1”作为分界符，将前缀前面的“0”或“1”与*分割开，然后将“*”用“0”来表示。例如：前缀 1101***，加入分割符后变成：11011***，用 0 代替*得：11011000。图 4 给出了验证数字 2 在范围[0，4]内的过程，给定的前缀 R 是 0***。

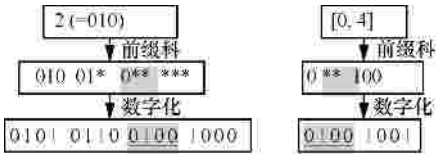

图 4　前缀成员确认

4.2　布鲁姆过滤器

前缀成员确认技术可将范围查询转换成成员确认。但为一个数字产生的前缀数据很多，直接存储这些前缀或者简单对这些前缀进行散列运算会产生两方面的问题。第一，空间消耗过大；第二，数据的真实值很容易通过查找的办法揭露出来。因此本文采用布鲁姆过滤器来存储这些前缀。

布鲁姆过滤器[24]是一种空间高效的随机化数据结构，用于元素集合的精简表示和隶属关系查询。标准布鲁姆过滤器采用长度为 m 的比特向量 V 以及 k 个相互独立的散列函数 h_1，h_2，…，h_k。当元素 s 存储到布鲁姆过滤器时，设置 V 中第 $h_1(s)$，$h_2(s)$，…，$h_k(s)$比特值为 1。当查询元素 u 是否在布鲁姆过滤器中时，检查 V 中第 $h_1(u)$，$h_2(u)$，…，$h_k(u)$比特值是否全为 1，如果全为 1，则元素 u 以较大概率在 S 中，如果不全为 1，则 u 不在布鲁姆过滤器中。

本文称具有相同长度比特向量且采用相同的 k 个散列函数的布鲁姆过滤器为同构布鲁姆过滤器。对于同构布鲁姆过滤器，定义n 运算。同构布鲁姆过滤器的n 运算定义为直接通过布鲁姆过滤器的比特向量的逻辑与运算完成，如图 5 所示。

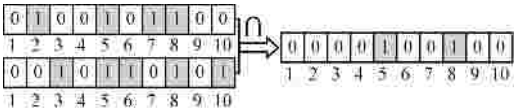

图 5　布鲁姆过滤器上的交运算

本文采用前缀成员确认技术和布鲁姆过滤器相结合的办法，在数据隐私的情况下判断数据 x 是否在指定的范围[a, b]内。其过程如下：给定前缀范围 R，为数据 x 产生 $F_R(x)$，并对每一个前缀采用单向散列函数 $H(\cdot)$ 计算其散列值，然后将散列结果采用布鲁姆过滤器存储。计算 $S([a, b])$，将 $S([a, b])$中的每一个前缀采用散列 $H(\cdot)$ 计算其散列值。并对每一个散列结果采用一个单独的布鲁姆过滤器进行存储，接下来判断对应 $S([a, b])$的所有布鲁姆过滤器中是否至少存在一个布鲁姆过滤器是存入 $F_R(x)$ 散列值后的布鲁姆过滤器的子集。如果存在，则 x 以很大概率在范围[a，b]内，否则不在该范围内。对验证数字 2 是否在范围[0，4]内，图 4 采用前缀成员确认技术产生了相关前缀。在此结果上，采用本节介绍布鲁姆过滤器的办法进行判断的过程如图 6 所示。

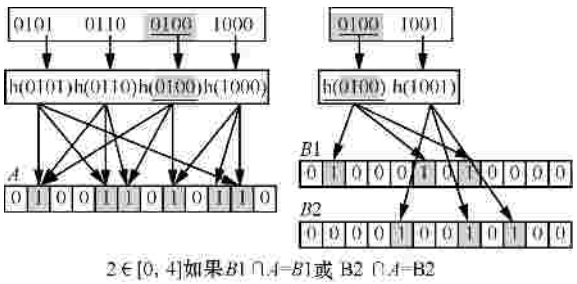

图 6　布鲁姆过滤器的成员确认

4.3　数据提交协议

数据提交协议处理传感器节点如何将采集的数据提交给存储节点。假设传感器节点 s_h 在一个周期中产生的数据为：$\{d_1, d_2, \mathrm{L}, d_n\}$，所有的数据都处在范围 R=[d_0，d_{n+1}]内，其中，d_0 和 d_{n+1} 分别代表数据的上届和下界。所有传感器和 sink 均知道范围 R。传感器 s_h 对收集的数据进行如下处理。

Step1　对每一个数据 d_j 计算其约束前缀科 $F_R(d_j)$，对前缀科中的数据进行散列后，为数据 d_j 构造一个布鲁姆过滤器 B_j，将该数据的约束前缀科的散列值存储在 B_j 中。

Step2　s_h 应用私钥 k_h 对所有数据进行加密，得到加密结果：$\{(d_1)_{k_h}, (d_2)_{k_h}, \mathrm{L}, (d_n)_{k_h}\}$。

Step3 s_h 将加密结果连同编码结果一起发送给附近的存储节点。

根据以上叙述，函数 $H(\cdot)$ 对应传感器 s_h 的第一步处理的散列方法，即为每一个数据分别计算带约束前缀集以及将该前缀集进行散列。

$X_h(\cdot)$ 传感器上的秘密布鲁姆过滤器中，由于每一个传感器采用不同的布鲁姆过滤器(包括散列表大小，散列函数个数以及不同的带密钥散列函数)，存储节点和妥协的传感器节点无法根据加密数据 $(d_j)_{k_h}$ 和编码结果 $X_h(H(d_j))$ 计算出 d_j。

4.4 查询协议

查询协议处理sink节点如何对查询条件进行编码，传感器节点如何对查询条件进行转换以及存储节点如何实现查询处理。

当需要执行范围查询$[a, b]$时，Sink、存储节点和传感器节点分别执行以下步骤来完成该查询任务。

Step1 Sink 计算范围$[a, b]$等价的最小前缀集 $S([a, b])$。

Step2 Sink 为前缀集 $S([a, b])$中的每一个前缀 p 采用共享散列函数计算对应的散列值，将进行散列的结果记作 $S_h([a, b])$。

Step4 Sink 将查询条件$\{t, S_h([a, b])\}$进行签名后发送给存储节点。

Step5 收到 sink 发送来的查询条件后，当存储节点要在传感器节点 s_h 提交的加密数据上执行查询时，存储节点首先将查询条件$\{t, S_h([a, b])\}$发送给传感器节点。s_h 首先对存储节点发送来的查询条件进行认证，确认查询条件是否由 sink 产生。当确认查询条件是由 sink 产生后，采用其上的秘密布鲁姆过滤器为每一个散列结果生成一个布鲁姆过滤器，得到新的查询条件$\{t, X_h(S_b([a,b]))\}$，将该查询条件发送给存储节点。

Step6 存储节点根据传感器节点发送来的查询条件执行查询，加密数据 $(d_j)_{k_h}$ 在查询结果中。当且仅当在布鲁姆过滤器集合 $X_h(S_b([a,b]))$ 中存在一个布鲁姆过滤器 B，使得 $X_h(H(d_j)) \mid B = B$ 成立。

Step7 存储节点将满足条件的所有加密数据进行聚合，并将聚合结果发送给 sink。

Sink 节点当中的秘密函数 $M(\cdot)$ 对应上述步骤的 Step1 和 Step2。由于散列函数的单向性，攻击者很难获得真实查询条件。存储节点上的查询函数对应 Step6 操作，即判断其中一个布鲁姆过滤器中置“1”的位置是否包含另外一个布鲁姆过滤器所有置“1”的位置。

5 基于代码表的数据压缩表示及传输

为了支持数据的隐私查询以及对各维的聚合运算，需要对数据不同维上的值进行单独编码和加密处理，产生了较大的加密和编码结果。因此需要研究如何减小传感器节点和存储节点之间的通信开销。对 Intel Lab[25]提供的真实数据 (该数据是 Intel Lab 在 2004 年 1 月 3 日至 2004 年 3 月 10 日期间部署的 44 个传感器采集的数据，并且每个传感器采集了温度、湿度与电压 3 种数据)进行了详细的统计分析，发现如果将这些多维数据作为一个整体看待，相邻周期数据的重复率很低。如取一个周期为 10min 时，相邻周期的三维数据的平均重复率只有 1.97%，二维数据的平均重复率也只有 4.52%。但是如果单独从某一维上的数据(如温度)看，相邻周期数据的重复率达到 59.4%；进一步地，如果保存前面 3~5 个周期的数据作为历史数据集，当前周期所产生的数据平均超过 99.5%的数据在历史数据集中，而且在随后的近 20 个周期中，每个周期平均都有超过 80%的数据在历史数据集中。

根据数据在同一维上重复的规律，本文数据压缩表示和传输的基本想法是：分别在传感器和存储节点上保存传感器已计算的数据的加密和编码结果，并用一个较短的代码来代替该数据。当传感器新采集的数据在代码表中存在时，传感器只需要在代码表中查询出该数据的代码发送给存储节点；存储节点获得传感器发送来的代码后，根据传感器 ID 以及发送来的代码在本地维护的代码表中查询出对应的加密和编码结果，从而避免传感器为该数据重复计算加密和编码结果以及减小传感器和存储节点之间的通信开销。

5.1 代码表设计

根据数据压缩表示的基本思想，本文分别为传感器节点和存储节点设计代码表。代码表的设计与维护应该满足以下条件：①代码表不能太大，以免占用过多的空间；②存储节点上的代码表和传感器节点上的代码表的同步不能过于频繁，以免消耗过多通信资源；③要尽可能地提高存储节点代码表的查询速度，为存储节点上的代码表存

储了单元内所有传感器提交的代码，传感器发送来的大部分数据需要在代码表中进行查询后得到数据的编码值和加密值，因此查询操作是代码表中最主要的操作。

为了满足以上条件，传感器上的代码表和存储节点上的代码表如表 1 和表 2 所示。其中，*Value*、*Code*、*Encode*、*Cipher*、*Sensor ID* 是一些基本属性，分别表示数据的真实值、代码值、编码结果、加密结果以及传感器节点 *ID*。传感器节点代码表中的 *Time Slot* 用于记录该代码最近使用的周期号，引入该属性是为了避免代码表随时间的推移而不断增大。当前周期与代码记录中的最近使用周期之差达到设定的阈值时，认为该代码已经过期；在传感器代码表中的可用空间不足时，批量地将过期代码记录从表中删除。传感器节点代码表中的 *Syn tag* 属性记录了该代码与存储节点的代码之间的同步情况，“1”表示已经同步，即在存储节点的代码表中有相应的记录；“0”表示没有同步，即存储节点上的代码表中还没有该记录。引入该属性是为了对代码进行批量更新，当传感器节点产生新数据时，只暂时更新本地代码表，并置该属性值为“0”，当达到一定条件后再向存储节点发起代码批量更新的请求。图 7 中 $E(d_i)$、$C(d_i)$ 和 $R(d_i)$ 分别表示对数据 d_i 的编码、加密和产生的随机代码的结果。

表 1　　$s_?$ 代码表

Value	Code	Encode	Cipher	Time Slot	Syn tag
d_i	$R(d_i)$	$E(d_i)$	$C(d_i)$	t_i	1
…					
d_j	$R(d_j)$	$E(d_j)$	$C(d_j)$	t_j	1

表 2　　存储节点代码表

Sensor ID	Code	Encode	Cipher
?	$R(d_i)$	$E(d_i)$	$C(d_i)$
…			
?	$R(d_j)$	$E(d_j)$	$D(d_j)$

5.2　数据表示与传输

本节讨论如何利用代码表进行数据的表示以及传输。设 $d_1, d_2, \perp, d_n$ 是传感器 s_h 在周期 t 中某一维上采集的数据，d_0 和 d_{n+1} 分别表示该维数据的上、下界。在周期结束时，传感器 s_h 进行如下两步操作。

Step 1　判断数据 d_i $(0 \le i \le n)$是否在代码表中。

1) d_i 在代码表中，更新该代码的最近使用周期为 t，并进一步判断 *Syn Tag* 的值。

Syn Tag 值为“1”：获得该数据的代码 $R(d_i)$ 作为向存储节点发送的数据。

Syn Tag 值为“0”：获得该数据的编码和加密结果 $E(d_i)$ 和 $C(d_i)$ 作为向存储节点发送的数据。

2) d_i 不在代码表中，分别计算该数据的编码和加密结果 $E(d_i)$、$C(d_i)$，并采用随机数发生器产生一个长度为 $\lfloor \mathrm{lb}(d_n - d_0) \rfloor$ 且与当前代码表中代码不重复的代码 $R(d_i)$ 作为该数据的代码，在本地代码表中插入记录（d_i, $R(d_i)$ $E(d_i)$ $C(d_i)$, t, 0）(如果代码表中的空间不足，在保存过期代码的前提下删除所有过期数据，然后进行数据插入)，并将 $E(d_i)$ 和 $C(d_i)$ 作为向存储节点发送的数据。

Step 2　传感器 $s_?$ 将通过第一步得到的发送数据发送给附近的存储节点。

存储节点接收到传感器 $s_?$ 发送来的各维上数据后，检查是否有代码数据，如果有代码数据，则根据传感器的 ID 号和数据代码在本地代码表中查找出相应数据的编码和加密结果。

6　分析

本节首先对提出方案的安全性进行分析，然后重点对协议的通信开销以及几个关键参数进行分析。

6.1　安全性分析

条件聚合协议的安全性包含两部分：加密数据的安全性以及编码数据的安全性。加密数据的安全性基于加密数方案本身，由于不同传感器节点采用不同的密钥对数据进行加密，因此攻击者在没有获得传感器密钥的情况下是很难破解传感器节点采集数据的真实值。在编码部分，每一个传感器分别采用了秘密编码布鲁姆过滤器，由于散列函数的单向性以及秘密性，因此攻击者无法根据编码结果解密数据。

6.2　几个重要参数的分析

接下来对编码结果的大小、假阳性以及同态加密数据的大小进行分析。

编码大小及假阳性分析：由于采用布鲁姆过滤器对数据进行编码，因此存在假阳性问题。对于要编码的数据，要知道其范围，即在范围 R 下的编码。例如，在环境检测中，在一个特定的地方，温度的

变化范围一般用8位就足以区分。假设在范围R下，数据可以用w位进行区分，则产生$n=w+1$个前缀，即一个布鲁姆过滤器中需要存储的元素个数。当选取布鲁姆过滤器的大小为 m 时，散列函数的个数$k=\ln 2 m/n$。根据假阳性的计算公式：$f \approx (1-e^{-km/n})^k \approx 0.6185^{m/n}$。在Intel Lab[25]提供的真实数据来验证本文的方案，用8位数据就能区分，即布鲁姆中需要存储的前缀数量是9，采用的布鲁姆过滤器的大小是91位，理论上数据的假阳性为0.819 2%。由于假阳性的存在，因此存储节点进行聚合的时候，聚合结果与真实值之间存在一定的偏差。设在某一维上数据的最大和最小值分别为$V_{\max}$和$V_{\min}$，假阳性比率为f，则在最差的情况下的偏差为：$f(V_{\max}-V_{\min})$。

同态加密数据的大小：由第2节介绍的同态加密原理知：同态加密结果长度为$l_h=\lfloor \mathrm{lb}n\times V_{\max}^2 \rfloor$。其中，$V_{\max}$为参加运算的数据的最大值，$n$为可以支持的参加运算的元素个数。$n$越大时，可以支持参加同态运算元素的个数越多，存储节点与sink节点的通信开销就越小；但n越大，同态加密的结果也就越大，存储节点和传感器节点的通信开销也就越大。因此需要综合考虑后来确定n的取值。设代码长度假为l_c，编码数据长度为l_e；传感器在一段时间内采集的数据的个数为M，这M个元素中通过代码上传的数据比率为r，同时有k个数据进行了代码更新，则传感器节点的通信大小为

$$S=M\times r\times l_c+(M\times(1-r)+k)\times(l_e+l_h) \quad (1)$$

式(1)中，代码长度、编码长度为常数，因此可以变化成：$S=(M\times(1-r)+k)\times\lfloor \mathrm{lb}n\times V_{\max}^2 \rfloor+C$。从式中可以看出：$n$越小，传感器发送的数据量就越小。假设在同一时间段内，需要对M'个数据进行聚合并发送给sink，则存储节点的通信空间开销为

$$S'=\lfloor M'/n \rfloor\times l_h=\lfloor M'/n \rfloor\times\lfloor \mathrm{lb}n\times V_{\max}^2 \rfloor \quad (2)$$

从式(2)中不难看出，n越大，存储节点发送给sink的开销就越小。在给定具体参数的情况下，可以求出n的最优值。

7 实验

本文算法中，能耗与文献[16]的能耗基本一致，即每发送32bit数据，需消耗33μJ左右的能量，为了便于分析本文算法在各个维度数据中的有效性，用传感器平均每一个周期向存数节点发送的数据量大小以及存储节点返给sink的平均每1个查询的结果大小来评估实验结果。在OMnet++平台上实现了本文提出的协议，并采用了Intel Lab[25]在2004年1月3日至2004年3月10日期间部署的44个传感器采集的数据。本文实验与SafeQ[12]采用了同样的网络拓扑结构图，即实验将44个传感器分成4个组，每个组内有11个传感节点和一个存储节点。随机产生了1 000个聚合查询对存储节点上的数据进行查询。

本文首先用实验的方法分析了本文协议中的几个重要参数：数据的重复率，不同阈值下数据的更新周期以及聚合数据n的取值。图7(a)给出在不同历史数据集大小下，数据重复率随时间变化关系。图中1、2、4、6、8分别表示将传感器在1个周期、2个周期、…、8个周期采集的数据作为历史数据的情况。从图中可以看出，历史数据集越大，数据重复率下降的越慢。图7(b)表示在不同阈值，不同历史数据集大小的情况下，数据平均更新周期。例如图7(b)中，当历史数据集为8时，设定更新阈值为80%(即当前周期数据与历史数据重复率不高于80%)，则平均90个周期需要更新一次历史数据集。图7(c)的n表示同态加密支持的数据聚合的个数。图中表示随n的不同，传感器和存储节点发送数据量大小的和的变化情况，图中n=66为最优。

图8是在确定同态加密支持66个数据进行聚合的情况下，采用聚合和不采用聚合情况下查询响应当中数据量的大小比较。本文聚合协议支持跨周期数据的聚合，当聚合中所跨周期越大，协议的优势越明显。计算参与聚合的周期数为1~8，对于一维数据，本文的协议发送数据量大小比不进行聚合平均要减小94.82%；二维数据要减小90.57%；三维数据减小82.76%。

不同更新阈值同样影响传感器节点发送数据量的大小，图9对比了一、二、三维数据在不同阈值下传感器发送数据量大小，为了更好地区分，图中曲线表示对前面周期数据发送量的总和，如*time slot*=5，表示前面5个周期传感器发送数据的总量，并且将更新数据量平均计入该总量中。

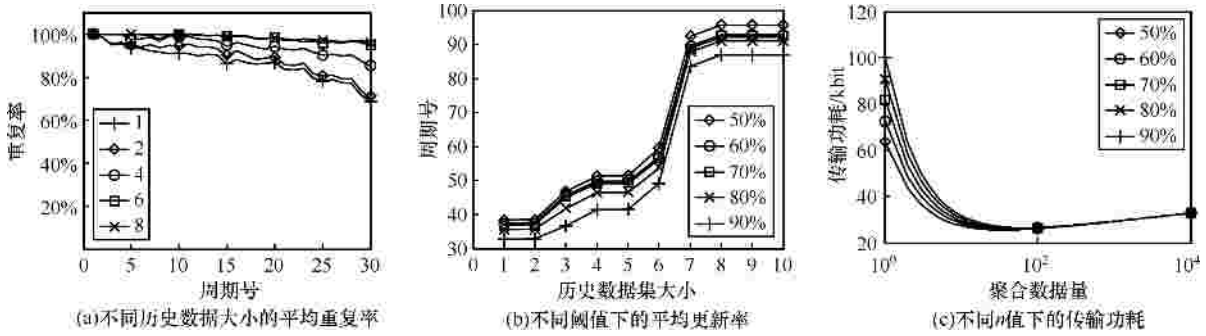

图 7　Intlab 数据中的几个重要参数

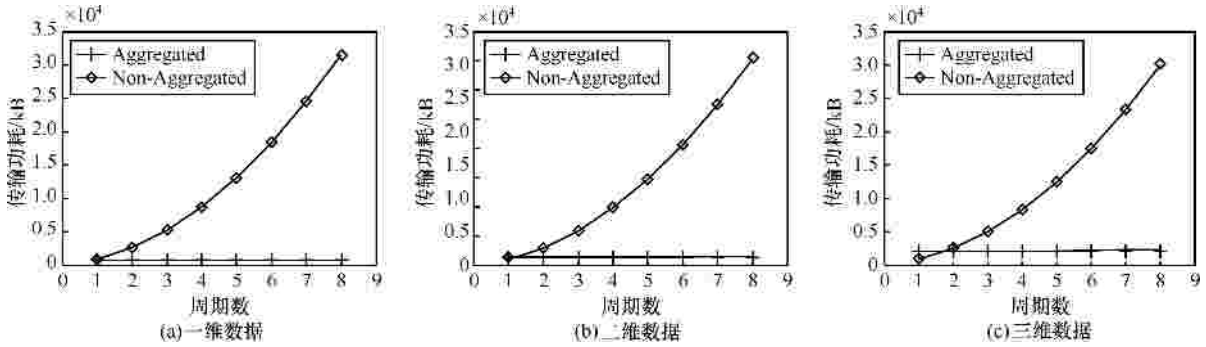

图 8　存储节点发送给 sink 的查询结果的平均大小

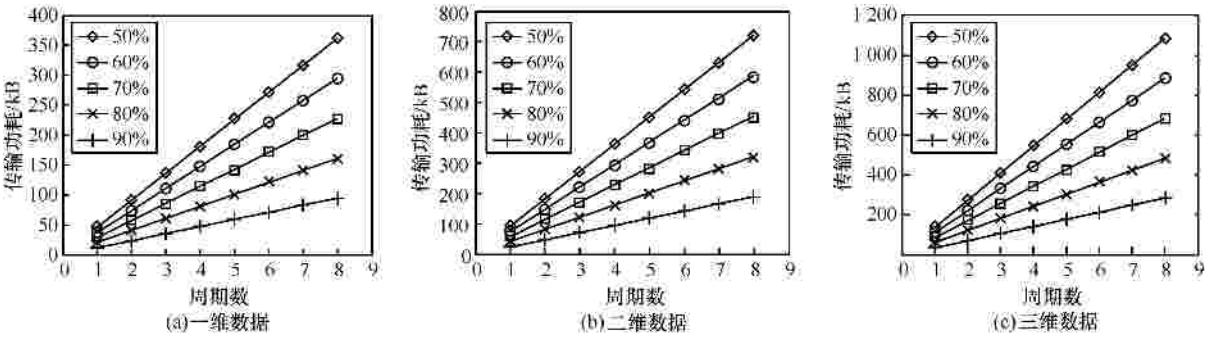

图 9　传感器在不同周期数的情况下提交给存储节点数据的平均大小

8　结束语

本文提出了一种隐私保护的条件聚合协议，使存储节点在不知道数据真实值的情况下对满足条件的数据进行聚合。为了保护数据和查询条件的隐私性，提出了一种基于前缀成员确认和布鲁姆过滤器相结合的编码方法，对数据和查询条件进行编码，实现存储节点在不知道数据真实值和查询条件真实值的情况下进行查询处理 为了对查询结果中的数据进行聚合而不暴露数据真实值，采用同态加密技术对数据进行加密，使数据在不解密的情况下能进行聚合运算。根据传感器采集数据的特点，提出了一种基于代码表的数据压缩表示以及传输方法，有效减小了传感器节点和存储节点之间的通信开销。分析和实验结果验证了所提方案的有效性。

参考文献：

[1] 李建中,李金宝,石胜飞. 传感器网络及其数据管理的概念、问题与进展[J]. 软件学报, 2003,14(10):1717-1727
LI J H, LI J B, SHI S F. Concepts, issues and advance of sensor networks and data management of sensor networks[J]. Journal of Software, 2003, 14(10):1717-1727.

[2] 崔莉,鞠海玲,苗勇等. 无线传感器网络研究进展[J]. 计算机研究与发展, 2005, 42(1):163-174.
CUI L, JU H L, MIAO Y, *et al*. Overview of wireless sensor networks[J]. Journal of Computer Research and Development, 2005, 42(1):163-174.

[3] DESNOYERS P, GANESAN D, LI H, SHENOY P. Presto: a predic-

tive storage architecture for sensor networks[A]. Proc 10th Workshop on Hot Topics in Operating Systems[C]. Berkeley: USENIX Association, 2005. 23-23.

[4] ZEINALIPOUR-YAZTI D, LIN S, KALOGERAKI V, *et al*. Microhash: An efficient index structure for flash-based sensor devices[A]. Proc 4th USENIX Conf. on File and Storage Technologies (FAST2005)[C]. 2005. 31-44.

[5] SHENG B, LI Q, MAO W. Data storage placement in sensor networks[A]. Proc 7th ACM Inte Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc2006)[C]. 2006. 344-355.

[6] SHENG B, TAN CC, LI Q, MAO W. An approximation algorithm for data storage placement in sensor networks[A]. Proc Inte Conf on Wireless Algorithms, Systems and Applications (Wasa 2007)[C]. 2007. 71-78.

[7] HEIDEMANN J. *et al*. Building efficient wireless sensor networks with low-level naming[A]. 18th ACM Symposium on Operating Systems Principles[C]. October 2001. 21-24.

[8] INTANAGONWIWAT C, GOVINDAN R, ESTRIN D. Directed diffusion: a scalable and robust communication paradig for sensor networks[A]. Proc Annual Inte Conf on Mobile Computing and Networking[C]. New York: ACM Press, 2000. 56-67.

[9] SHENG B, LI Q. Verifiable privacy-preserving range query in two-tiered sensor networks[A]. Proc IEEE Inte Conf on Computer Communications (INFOCOM2008)[C]. 2008. 46-50.

[10] SHI J, ZHANG R, ZHANG Y. Secure range queries in tiered sensor networks[A]. Proc IEEE Inte Conf on Computer Communications (INFOCOM2009)[C]. 2009. 945-953.

[11] ZHANG R, SHI J, ZHANG Y. Secure multidimensional range queries in sensor networks[A]. Proc ACM Inte Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc2009)[C]. 2009. 197-206.

[12] CHEN F, LIU A X. SafeQ: Secure and efficient query processing in sensor networks[A]. Proc IEEE Inte Conf on Computer Communications (INFOCOM 2010)[C]. 2010. 2642-2650.

[13] PRZYDATEK B, SONG D, PERRIG A. SIA: secure information aggregation in sensor networks[A]. Proc Inte Embedded Networked Sensor Systems (SenSys'03)[C]. 2003. 255-265.

[14] WANG C, WANG G, ZHANG W, FENG T. Reconciling privacy preservation and intrusion detection in sensory data aggregation[A]. Proc IEEE Inte Conf on Computer Communications (INFOCOM 2011)[C]. 2011. 336-340.

[15] CASTELLUCCIA C, MYKLETUN E, TSUDIK G. Efficient aggregation of encrypted data in wireless sensor networks[A]. Proc Annual Inte Conf on Mobile and Ubiquitous Systems: Networking and Services[C], San Diego, CA, USA, 2005. 109-117.

[16] YAO Y, XIONG N, PARK J, *et al*. Privacy-preserving max/min query in two-tiered wireless sensor networks[J]. Computer and Mathematics with Application. 2012, 2:1-8.

[17] GOLDREICH O. Foundations of Cryptography: Vol. 2, Basic Applications[M]. Cambridge University Press, New York, NY, USA, 2004.

[18] CHAN H, PERRIG A, SONG D. Secure hierarchical in-network aggregation in sensor networks[A]. Proceedings of the 13th ACM conference on Computer and communications security[C]. New York: ACM Press. 2006: 278 - 287.

[19] YANG Y, WANG X. ZHU S, *et al*. Sdap: a secure hop-by-hop data aggregation protocol for sensor networks[J]. ACM Transactions on Information and System Security, 2008, 11(4): 1-43.

[20] ISKANDER M K, Lee A J, Mossé D. Privacy and robustness for data aggregation in wireless sensor networks[A]. Proceedings of the 17th ACM Conference on Computer and Communications Security[C]. New York: ACM Press, 2010. 699-701

[21] CHENG J, YANG H, WONG S H, *et al*. Design and implementation of cross-domain cooperative firewall[A]. Proc. International Conference on Network Protocols[C]. Piscataway: IEEE, 2007. 284-293.

[22] LIU A X, CHEN F. Collaborative enforcement of firewall policies in virtual private networks[A]. Proceedings of the Twenty-Seventh ACM Symposium on Principles of Distributed Computing[C]. New York: ACM Press, 2008. 95-104.

[23] CHANG Y K. Fast binary and multiway prefix searches for packet forwarding[J]. Computer Networks, 2007, 51(3):588-605.

[24] BRODER A, MITZENMACHER M. Network applications of Bloom filters: A survey[J]. Internet Mathematics, 2004, 11(4): 485-509.

[25] Intel lab data[EB/OL]. http://berkeley.intel-research.net/ labdata.

作者简介：

李睿（1975-），男，湖南汨罗人，博士，湖南大学讲师，主要研究方向为无线传感器网络、网络安全。

林亚平（1955-），男，湖南邵阳人，博士，湖南大学教授、博士生导师，主要研究方向为无线传感器网络。

李晋国（1985-），男，湖南衡阳人，湖南大学博士生，主要研究方向为两层无线传感器网络。