南京邮电大学

毕 业 设 计（论 文）

|  |  |
| --- | --- |
| 题 目 | 一种改进的两层传感器网络隐私保护最值查询方法 |
| 专 业 | 软件工程（NIIT） |
| 学生姓名 | 戴 铭 洋 |
| 班级学号 | B10041234 |
| 指导教师 | 戴 华 |
| 指导单位 | 南京邮电大学 |

日期： 2014 年 3 月 10 日至 2014 年 6 月 13 日

毕业设计(论文)原创性声明

本人郑重声明：所提交的毕业设计(论文)，是本人在导师指导下，独立进行研究工作所取得的成果。除文中已注明引用的内容外，本毕业设计(论文)不包含任何其他个人或集体已经发表或撰写过的作品成果。对本研究做出过重要贡献的个人和集体，均已在文中以明确方式标明并表示了谢意。

论文作者签名：

日期： 年 月 日

摘 要

无线传感器网络已被应用于很多重要的领域并在很多潜在的领域有广泛的应用前景。为了进一步降低现有两层无线传感器网络隐私保护最值查询处理方法的感知节点能耗，本毕设提出了一种改进的基于0-1编码的两层无线传感器网络隐私保护最值查询处理协议，并根据协议设计和实现两层无线传感器网络隐私保护最值查询模拟系统。

(1) 研究了无线传感器网络数据查询的原理和方法，介绍了现有的无线传感器网络隐私保护技术的相关工作；

(2) 介绍了安全多方计算中的0-1编码技术、Hash身份认证编码技术和当前适用于两层传感器网络环境的感知数据秘密比较机制；

(3) 结合两层无线传感器网络特点和秘密比较原理，给出了实现改进的两层无线传感器网络隐私保护最值查询的数据上传协议和查询处理协议，仿真实验表明，该协议与现有协议相比具有更高的能耗效率，从而提高了整个网络的使用寿命；

(4) 根据上述相关理论基础，设计并实现了两层无线传感器网络隐私保护最值查询模拟系统，该系统给出了网络中感知节点的数据收集、Sink节点发出查询请求过程以及存储节点执行隐私保护最值查询的执行处理过程。

关键词：两层无线传感网；隐私保护；最值查询；0-1编码

**ABSTRACT**

Wireless sensor network (WSN) has been applied in many important areas and have broad application prospects in many potential field. In order to reduce the energy consumption of sensor nodes in the existing privacy-preserving MAX/MIN query protocol, this graduation project proposes an improved 0-1 encoding based privacy-preserving MAX/MIN query protocol and implements a MAX/MIN query simulation system.

(1) The principles and methods of data query in WSN have been studied and some protocols of privacy protection of the sensor network have been introduced.

(2) The 0-1 encoding technique of secure multi-party computation, hash authentication coding technique and a secret data comparison mechanism for Two-tiered WSNs have been introduced.

(3) Combine Two-tiered WSNs characteristics and secret comparison mechanism, we propose the data submit protocol and secure comparison protocol. And the simulation results show that, the improved protocol has better energy efficiency than the existing protocol, thereby the lifetime of the whole network is prolonged.

(4) According to the protocol, we design and implement an improved two-tiered WSNs privacy-preserving MAX/MIN query simulation system, which describes how the sensor nodes collect data, the Sink node issue queries and the storage node performs the query execution of privacy protection.

**Key words**：Two-tiered wireless sensor networks；Privacy-preserving；MAX/MIN query；0-1 encoding

目 录

[第一章 绪论 1](#_Toc389826169)

[1.1 研究背景 1](#_Toc389826170)

[1.2 无线传感器网络隐私保护技术 2](#_Toc389826171)

[1.3 研究内容及安排 4](#_Toc389826172)

[第二章 相关工作 5](#_Toc389826173)

[2.1 隐私保护数据聚集技术 5](#_Toc389826174)

[2.2 隐私保护范围查询 5](#_Toc389826175)

[2.3 隐私保护Top-*k*查询 6](#_Toc389826176)

[2.4 隐私保护最值查询 7](#_Toc389826177)

[2.5 本章小结 7](#_Toc389826178)

[第三章 隐私保护关键技术 8](#_Toc389826179)

[3.1 0-1编码 8](#_Toc389826180)

[3.1.1 0-1编码的定义 8](#_Toc389826181)

[3.1.2 0-1编码的性质和定理 8](#_Toc389826182)

[3.1.3 0-1编码的数值化 10](#_Toc389826183)

[3.2 HMAC算法 11](#_Toc389826184)

[3.2.1 散列函数 11](#_Toc389826185)

[3.2.2 HMAC算法的定义 12](#_Toc389826186)

[3.2.3 HMAC安全性分析 13](#_Toc389826187)

[3.3 对称加密技术 14](#_Toc389826188)

[3.3.1 DES基本概念 14](#_Toc389826189)

[3.3.2 DES算法工作原理 15](#_Toc389826190)

[3.4 本章小结 17](#_Toc389826191)

[第四章 改进的隐私保护最值查询协议 18](#_Toc389826192)

[4.1 威胁模型与协议模型 18](#_Toc389826193)

[4.1.1 威胁模型与问题描述 18](#_Toc389826194)

[4.1.2 协议模型 18](#_Toc389826195)

[4.2 数据上传协议 19](#_Toc389826196)

[4.3 查询处理协议 20](#_Toc389826197)

[4.4 协议分析 22](#_Toc389826198)

[4.4.1 隐私安全性分析 22](#_Toc389826199)

[4.4.2 能耗分析 22](#_Toc389826200)

[4.5 实验分析 22](#_Toc389826201)

[4.6 本章小结 26](#_Toc389826202)

[第五章 OMQP模拟系统的设计与实现 27](#_Toc389826203)

[5.1 模拟系统设计框架 27](#_Toc389826204)

[5.1.1 模拟系统界面设计 27](#_Toc389826205)

[5.1.2 模拟系统设计背景和数据结构设计 28](#_Toc389826206)

[5.2 主要函数的设计与实现 28](#_Toc389826207)

[5.2.1 0-1编码函数实现 28](#_Toc389826208)

[5.2.2 定时器事件函数实现 30](#_Toc389826209)

[5.2.3 最值密文候选集生成算法 31](#_Toc389826210)

[5.2.4 其他函数的实现 33](#_Toc389826211)

[5.3 模拟系统的工作过程 34](#_Toc389826212)

[5.4 本章小结 35](#_Toc389826213)

[结束语 36](#_Toc389826214)

[致 谢 37](#_Toc389826215)

[参考文献 38](#_Toc389826216)

# 第一章 绪论

## 1.1 研究背景

无线传感器网络(Wireless Sensor Networks，WSN)是近几年来迅速发展并得到普遍重视的新型网络技术，它的出现和发展对现代的科学技术造成了十分深刻的影响。与传统网络技术不同，无线传感器网络技术将现代的网络技术、微型传感器技术和无线通信技术有效的融为一体，成为近几年来国内外的研究热点，引起了许多国家学术界、工业界和军界的高度重视，其应用前景十分的广阔。

WSN是由大量密集分布在监控区域的具有无线通信能力与计算能力的感知节点构成的一种网络应用系统。WSN通过无线通信的方式形成一个多跳的自组织(Ad hoc)的网络系统，感知、采集和处理监控区域中目标的信息，并将信息发送给观察者，被广泛应用于工业应用、交通控制、设施安全、环境监控等。

与传统无线Ad hoc网络不同，WSN的节点没有统一的标识，节点之间通过广播、多跳的通信方式交换数据。WSN的节点数量很多，随机分布，网络拓扑结构随着时间动态变化，节点各供电电源能量有限，生命周期很短。所以WSN需要研究新的技术，以实现网络中能量消耗最小化、节点的生命周期最大化、能量负载的均衡化和通信能力最优化的目标。

两层传感器网络(Two-tiered Wireless Sensor Networks，两层WSNs)是一种由感知节点（Sensors Node)、存储节点(Storage Node)、汇聚节点(Sink)组成的无线传感器网络（如图1.1所示）。在两层传感器网络中，整个网络被分割为多个单元，每个单元包含一个存储节点和多个感知节点，感知节点是大量分散部署廉价的传感器，低廉的价格导致其内存空间和计算能力都很有限，主要用于收集附近的自然数据，如温度等；存储节点是有更多的存储空间和强大的计算能力的移动设备，用于收集工作范围内的感知节点获得的数据，并且能够用其距离远、频率高的通信能力与其相邻存储节点进行通信，形成多跳的上层通信网络(Upper-tire Network)；汇聚节点是能够与用户直接交流的设备，它与存储节点之间使用按需无线链接(On-demand Wireless Link)的方式进行数据通信，例如使用卫星通信，这种无线链接通常具有不稳定、代价高和速率低等特点。这就需要查询处理尽量在存储节点内进行，查询精度越高越好，以尽量降低存储节点与汇聚节点之间的通信代价。每当汇聚节点收到来自用户的请求，它根据请求内容把查询请求发送给有关的存储节点，存储节点查询请求内容并将查询结果发送给汇聚节点，最后汇聚节点整合各个存储节点发送来的查询结果并返回给用户。



图1.1 两层传感器网络体系结构图

在两层传感器网络中，存储节点为附近的感知节点提供数据存储服务，同时负责执行汇聚节点发送来的查询请求。相对于传统无线传感器网络，两层传感器网络的优势主要有三点：（1）感知节点将所有收集到的数据发送到离他们最近的存储节点而不是通过较长的路径发送给汇聚节点，从而降低了传感器通信消耗；（2）由于传感器周期性将数据传送给存储节点，改善了传感器存储资源不足的问题，使得小内存传感器的应用成为可能；（3）由于数据存储在存储节点，执行查询时只需在存储节点执行，查询的处理过程将变得更加高效。

由于两层传感器网络的拓扑结构的简单性和存储节点的资源充裕性，使得两层传感器网络具有链路质量稳定、路由结构简单、查询高效和负载均衡等优点。然而，由于存储节点不仅存储着大量感知数据，而且还负责执行上层的查询请求，攻击者往往选择存储节点作为其首要攻击目标。若存储节点被俘获，攻击者可能利用被俘获的存储节点窃取大量的感知数据信息和查询处理结果，对整个传感器网络的数据隐私产生严重威胁。因此，如何设计一个安全有效的查询协议已成为主要的研究目标。

## 1.2 无线传感器网络隐私保护技术

传感器网络是物联网的一个重要组成部分，在很多领域都有广阔的应用前景。随着传感器网络的应用发展，它在实际应用过程面临严重的隐私数据泄漏或数据被篡改的问题。在医疗应用领域，便携式无线传感器收集患者或体检者的心率、血压等重要的敏感数据可能被泄漏；在家居领域，攻击者可能会根据用电、用水等数据推测居民是否在家等日常生活情况；在军事领域，无线传感器网络收集和查询的携带重要情报信息的数据，如果数据泄漏将带来严重后果。因此，解决传感器网络中的数据隐私保护问题，对传感器网络的大规模推广应用有重要的意义。

传感器网络具备多跳、自组织、分布式、资源受限和以数据为中心等特征，且往往部署在无人看守、不易控制的复杂环境中。传感器网络中数据隐私保护主要面临以下挑战：(1) 部署环境不容易控制：由于传感器网络往往部署在无人看守的环境中，攻击者除了可能通过链路层窃听获取敏感信息外，还可能通过俘获控制或伪造感知节点进行窃取或篡改敏感信息。(2) 资源受限：资源受限是传感器网络的重要特征，能量的消耗量直接影响传感器网络的寿命，这使得适用于传统网络中数据采集或数据发布的隐私保护技术不能直接用于传感器网络。(3) 网络形态的多样性：传感器网络能够用于监测、医疗、军事等领域，其网络形态和特征不同，可能存在不同的攻击模型和隐私保护需求，需要有针对性地设计隐私保护协议。

传感器网络中的隐私保护技术主要是数据隐私保护技术和位置隐私保护技术。本毕设重点研究数据隐私保护技术，该技术主要为了实现在不泄露隐私信息的情况下完成数据聚集、数据查询和访问控制等任务，并进行性能优化以减少能量消耗、时间延迟和数据丢失率。

根据传感器网络数据操作任务的不同，隐私保护协议针对的网络模型、攻击模型和安全目标也往往不同。同时，隐私保护协议采用的隐私保护技术和优化策略不同，其隐私保护能力、算法性能和结果精确度等性能指标也存在较大差异。

表1.1给出了各类操作任务及其实现技术分类情况。

表1.1 传感器网络隐私保护技术研究分类

|  |  |  |  |
| --- | --- | --- | --- |
| 操作任务 | | 主要技术 | 代表协议 |
| 隐私保护数据聚集 | | 逐跳加密机制 | CPDA[7]、DADPP[8]、SMART[7]、  iPDA[9]、ESPART[10] |
| 端到端  加密机制 | AHE[11]、SP[12]、SIES[13]、CDA[14]、IPHCDA[15] |
| 非加密策略 | KIPDA[16]、GP2S[17] |
| 隐私保护  数据查询 | 范围查询 | 桶模式 | Sheng&Li方案[18]、Shi方案[19]、  Zhang方案[20] |
| 前缀成员验证 | SafeQ[21] |
| Top-*k*查询 | 扰动和安全  比较技术 | SafeTQ[22] |
| 基于类型  查询 | 椭圆曲线多项式转换技术 | ElliPS[23] |
| 隐私访问控制 | | 盲签名 | DP2AC[24] |
| 环签名 | Priccess[25] |

## 1.3 研究内容及安排

本毕设主要研究基于0-1编码的安全比较技术。本毕设的主要工作包括：

1. 研究了无线传感器网络数据查询的原理和方法，介绍了现有的无线传感器网络隐私保护技术的相关工作；
2. 介绍了安全多方计算中的0-1编码技术、Hash身份认证编码技术和当前适用于两层传感器网络环境的感知数据秘密比较机制；
3. 结合两层无线传感器网络特点和秘密比较原理，给出了实现改进的两层无线传感器网络隐私保护最值查询的数据上传协议和查询处理协议，仿真实验表明，该协议与现有协议相比具有更高的能耗效率，从而提高了整个网络的使用寿命；
4. 根据上述相关理论基础，设计并实现了两层无线传感器网络隐私保护最值查询模拟系统，该系统给出了网络中感知节点的数据收集、Sink节点发出查询请求过程以及存储节点执行隐私保护最值查询的执行处理过程。

本毕设主要包括五个部分，论文的组织结构如下：

第一章介绍了传统无线传感器网络和两层无线传感器网络以及本毕设的研究意义，同时列举了目前无线传感器网络隐私保护技术，给出了本毕设的主要研究内容和安排；

第二章介绍了隐私保护的数据聚集技术、范围查询协议、隐私保护Top-*k*查询和隐私保护最值查询；

第三章介绍了用于实现秘密比较机制的关键技术，包含0-1编码、Hash消息身份认证以及对称加密；

第四章给出了两层传感器网络中的安全威胁模型，给出了实现改进的两层无线传感器网络隐私保护最值查询的数据上传协议和查询处理协议，同时，对协议进行安全性和能耗分析，并给出与现有技术的实验对比分析；

第五章设计并实现了改进的两层传感器网络隐私保护最值查询模拟演示程序，对隐私保护最值查询的工作过程进行模拟；

最后对全文进行总结和展望。

# 第二章 相关工作

## 2.1 隐私保护数据聚集技术

传感器网络数据聚集通过在聚集节点进行数据融合或压缩，能够有效地减少网络通信量，是传感器网络中重要的减少能量消耗技术。由于聚集节点需要收集聚集数据，容易成为攻击者攻击的对象。攻击者俘获控制聚集节点后，一方面能够获知经过聚集节点的明文数据，另一方面能够获得聚集节点密钥，能够对采用逐跳加密机制加密的数据进行解密，从而窃取或篡改所有经过聚集节点的敏感数据。隐私保护数据聚集的主要设计目标为在聚集节点不能获知所收集感知数据的情况下计算出聚集结果，在采用感知节点和Sink节点共享密钥的端到端加密机制时，因为聚集节点不能进行解密，在聚集节点容易实现隐私保护，但该技术需要在加密数据上实现数据聚集。

综合现有传感器网络隐私保护数据聚集研究成果，主要有以下几种实现策略：CPDA(Cluster-based Privacy Data Aggregation)[7]中感知节点通过在原始数据中添加随机种子和私有随机数进行扰动处理来隐藏真实数据值，簇头节点利用多项式的代数性质求解出精确的SUM聚集结果。DADPP(Data Aggregation Different Privacylevels Protection)[8]能够提供不同隐私保护水平的聚集处理，其处理过程类似于CPDA。SMART(Slice-Mixed AggRegaTion)[7]使用切分重组技术完成隐私保护数据聚集。iPDA(Integrity-Protecting Private Data Aggregation)[9]使用SMART中数据切分重组思路实现数据聚集中隐私保护，并通过冗余数据对聚集结果进行完整性验证。ESPART (Energy-saving Private-preserving AggRegaTion)[10]对SMART进行了改进，在节省能量消耗的情况下，实现了隐私保护数据聚集。AHE(Addictively Homomorhic Encryption)[11]使用一种简单的求和同态加密函数，实现采用端到端加密机制的SUM聚集。CDA (Concealed Data Aggregation)[12]采用Domingo-Ferrer提出的同态加密模式(DF scheme)实现传感器网络中隐私保护数据聚集。

## 2.2 隐私保护范围查询

Sheng&Li的方案[18]首次考虑传感器网络范围查询中隐私保护问题，采用桶模式(Bucketing Scheme)[26-27]和加密技术完成隐私保护范围查询，通过添加验证编码进行完整性验证。基本思路为：感知节点和Sink节点保持同样的桶划分，感知节点将采集的数据映射到相应的桶中，使用与Sink节点共享的密钥按桶分别进行加密，附加桶标签(Tag)后上传至存储节点进行存储，Sink节点将范围查询对应的标签发给存储节点，存储节点进行标签匹配后，将相应加密数据返回，Sink节点解密后得查询结果。

Shi等人的方案[19]同样采用桶模式完成隐私保护范围查询，其主要贡献为提出时空交叉验证方法(Spatiotemporal Crosscheck Approach)对查询结果进行验证。时空交叉验证方法由空间交叉验证技术和时间交叉验证技术组成，空间交叉验证技术主要思路为：感知节点生成表明采集数据在桶中分布情况的数据索引，在数据提交前广播该数据索引，其它节点收到后将数据索引嵌入本节点上传数据中，Sink节点利用数据索引进行查询结果完整性验证。时间交叉验证技术在上传数据中嵌入本节点表明不同时间段数据分布的数据索引，使用此数据索引可以对存储节点删除某时间段内全部数据的行为进行检测。Zhang等人的方案[20]对时空交叉验证方法进行了扩展，使用概率技术平衡隐私保护和能量消耗，并将方法应用于多维数据范围查。SafeQ(Secure and Efficient Query)[21]使用前缀成员验证 (Prefix Membership Verification，PMV) 技术在编码数据上实现范围查询而不泄露数据值，使用加密数据链技术完成完整性验证。

## 2.3 隐私保护Top-*k*查询

SafeTQ(Safe Top-*k* Query)[22]采用加随机数扰动和安全比较等技术在两层传感器网络中完成隐私保护精确Top-*k*查询，使用两种完整性验证模式使Sink节点能够检测和拒绝不正确或不完整查询响应。SafeTQ基于与图1-3相似的网络模型，传感器网络络被划分为若干个单元(cell)，单元划分时应保证每个单元区域至少存在2个高资源节点，选择单元内一个高资源节点作为单元头结点，另外指定一个高资源节点作为辅助计算节点。SafeTQ基于的查询模型为：

*Qt* = (cell = *C*)(query region = *G*)(epoch = *t*)(num = *k*)

其中，*C*为查询单元，*G*为查询区域，*t*为查询周期，*k*为查询需要返回的最大(或最小)的数据项数。SafeTQ仅描述查询区域在同一单元内的情况，对于查询区域为多个单元的情况能够进行分解处理。

图2.1给出了SafeTQ基本步骤。



图2.1 SafeTQ基本步骤演示

## 2.4 隐私保护最值查询

Yao等人提到尽管世界在研究隐私保护范围查询和隐私保护Top-*k*查询，隐私保护最值查询仍没有得到很好的解决。因此，他们提出了一个隐私保护最值查询协议阻止攻击者从传感器收集的数据中获得有益的信息。他们使用前缀成员验证（Prefix Membership Verification，PMV）的方法加密感知数据，使得存储节点在不知道真实值的情况下处理数据。

PMV在文献[1]中被首次提出，在文献[2]中首次给出定义。其主要思想是将判断数据是否在某个范围转换成比较数据的大小。由*k*个0或1作前项，*w-k*个\*作后项组成的编码序列叫做*k-*前缀编码，形如{0,1}*k*{\*}*w-k*。例如，110\*\*可以表示[11000,11011]中任意的二进制数。给出一个*w*位的二进制数*b*1*b*2…*bw* ，其前缀编码共有*w*+1个，为{*b*1*b*2…*bw*，*b*1*b*2…*bw-1*\*，…，*b*1\*…\*，\*…\*}，前缀编码集合记为*F*(*x*)。给出一个区间[*a* ,*b*]（假设*a* ,*b*都是*w*位的二进制数），其前缀编码集合记为*S*([*a* ,*b*])= {*p*1，*p*2，…，*pi*}。 *p*1，*p*2，…，*pi*为前缀编码，且其对应区间的并集等于[*a* ,*b*]，*p*1，*p*2，…，*pi*是满足前一条件的最小集合。PMV可以判断数字是否在某一区间内，当且仅当*F*(*x*)∩*S*([*a* ,*b*])，*x*[*a* ,*b*]。例如，*F*(14)∩*S*([8,15])= {1\*\*\*}，能推出14在[8,15]区间内。然而前缀编码机制产生的数据量较大，并且每一个感知节点都需要加密感知数据并传送密文数据，导致该查询处理过程需要较大的能耗开销。

Dai等人在文献[38]中提出一种节能的隐私保护最值查询协议（EMQP）。该协议通过使入0-1编码技术、Hash消息身份验证编码机制和对称加密技术，对感知数据进行编码处理，然后由感知节点将编码数据发送至存储节点，感知节点根据存储节点需求计算并传送加密数据；存储节点利用0-1 编码的数值比较特性，实现在无需感知数据明文参与下的数值线性关系比较，进而构造局部查询结果并发送给Sink，由Sink完成最终的最值查询结果计算。

该方法使用Hash消息身份验证编码机制，有效的减少了感知节点的能耗。本毕设提出的改进的算法能够进一步减少感知节点传送的数据，从而达到减少能耗的目标。

## 2.5 本章小结

本章简单介绍了隐私保护范围查询、隐私保护Top-*k*查询和隐私最值查询，同时对现有查询技术进行了研究和分析，指出了现有技术的优点和需要改进的地方。

# 第三章 隐私保护关键技术

## 3.1 0-1编码

0-1编码(Zero-One Encoding)是一种数值比较的编码方案，主要是对数字进行0编码和1编码，然后通过比较不同数字的0编码集合和1编码集合的集合关系，来确定数字间的大小关系。本毕设所提出的改进的两层传感器网络隐私保护最值查询方法，用到了基于0-1编码的秘密比较。

### 3.1.1 0-1编码的定义

设包含*w*位的二进制数值*x = b*1*b*2*…bw*-1*bw* (*bi*{0, 1}且1≤ *i* ≤*w*)。我们对*x*进行0-1编码，得到*x*的0编码集合*E0*(*x*)和1编码集合*E1*(*x*)。

0编码集合产生方案：设有*x = b*1*b*2*…bw*-1*bw* (*bi*{0, 1}且1≤ *i* ≤*w*)，*x*的0编码集合*E0*(*x*)初始情况下为空集。从*x*的最高位*b*1开始到最低位*bw*结束，对每一个二进制位*bi*进行判定，若*bi* 为“0”，则将二进制序列*b*1*b*2*…bi-*11加入到集合*E0*(*x*)中。0编码集合的产生过程可用以下公式表示：

*E0*(*x*) = {*b*1*b*2*…bi-*11 | *bi* = 0  1 ≤ *i ≤w*}

1编码集合产生方案：设有*x = b*1*b*2*…bw*-1*bw* (*bi*{0, 1}且1≤ *i* ≤*w*)，*x*的1编码集合*E1*(*x*)初始情况下为空集。从*x*的最高位*b*1开始到最低位*bw*结束，对每一个二进制位*bi*进行判定，若*bi* 为“1”，则将二进制序列*b*1*b*2*…bi*加入到集合*E1*(*x*)中。1编码集合的产生过程可用以下公式表示：

*E1*(*x*) = {*b*1*b*2*…bi* | *bi* = 1  1 ≤ *i ≤w*}

### 3.1.2 0-1编码的性质和定理

**性质3.1：**设数值*x*为包含*w*个二进制位，则0编码集合*E0*(*x*)和1编码集合*E1*(*x*)具有如下特点(设|X|表示集合X中的元素数量)：

1. *b*1*b*2*…bi*∈*E0*(*x*)∪*E1*(*x*)*bi*= 1
2. *E0*(*x*)∩*E1*(*x*)=
3. | *E0*(*x*)| + | *E1*(*x*)| = *w*

(1)证明：由0-1编码规则可知，*E0*(*x*)和*E1*(*x*)中任一元素的最后一位均为1。因此，(1)成立。

(2)证明：*E0*(*x*)末尾数原本是0，*E1*(*x*)末尾数原本是1，一个二进制数的*E0*(*x*)和*E1*(*x*)不存在两个相同位数的元素，也就不存在交集，性质(2)成立。

(3)证明：由0-1编码规则可知，*E0*(*x*)的元素数量与*x*中1的个数相等，而*E1*(*x*)的元素数量与*x*中0的个数相等，*E0*(*x*)和*E1*(*x*)中的元素数量之和为*x*的二进制编码总位数*w*。

**性质3.2：**若已知数值*x*的0编码或1编码集合，可以反向逆推出数值*x*。

由0-1编码方法及性质3.1的内容，易得性质3.2成立。例如，若已知包含5个二进制位的数值*x*的0编码集合*E0*(*x*)={1}，则*x*必定为01111。

**定理3.1：**对于都包含*w*个二进制位数值*x*和*y*，当且仅当*E1*(*x*)与*E0*(*y*)不相交时，*x* > *y*成立；当且仅当*E1*(*x*)与*E0*(*y*)存在非空交集时，*x* ≤ *y*成立，即有式3-1和式3-2成立。

*x > y E1*(*x*)∩*E0*(*y*)≠ (式3-1)

*x ≤ y E1*(*x*)∩*E0*(*y*)= (式3-2)

**证明：**设二进制*x*和*y*分别为*x*=*b*1*b*2*…bw*-1*bw*和*y*=*c*1*c*2*…cw*-1*cw*，我们首先证明式3-1的正确性，然后再根据式3-1的结论，用反证法证明式3-2的正确性。

式3-1充分性证明：设*x* > *y*成立，则在*x*和*y*的二进制编码中，必定存在第*i*位*bi*和*ci*满足(1≤ *i* ≤ *w*)：*bi*=1且*ci*=0成立，同时，对于第*i*位之前的任意一位*bj*和*cj*(1≤ *j* ≤*i*-1)，*bj*=*cj*也成立，即*b*1*b*2*…bi*-1=*c*1*c*2*…ci*-1成立。根据0编码方法可知*c*1*c*2*…ci*-11 *E0*(*y*)，同理，根据1编码方法易得*b*1*b*2*…bi*-11 *E1*(*x*)，而*b*1*b*2*…bi*-1=*c*1*c*2*…ci*-1，因此*E1*(*x*)∩*E0*(*y*) ≠成立。

式3-1必要性证明：假设*E1*(*x*)∩*E0*(*y*)≠成立，不妨设二进制序列*a*1*a*2*…ai**E1*(*x*)∩*E0*(*y*)，其中1≤ *i* ≤*w*，由性质1-(1)可知，*ai*=1成立，因此*a*1*a*2*…ai*-11*E1*(*x*)∩*E0*(*y*)成立。根据1编码方法易得，*b*1*b*2*…bi*-1*bi*=*a*1*a*2*…ai*-11成立，故*x*的二进制编码的高*i*位为*a*1*a*2*…ai*-11；同理，由0编码方法可知，*c*1*c*2*…ci*-1*ci*=*a*1*a*2*…ai*-10成立，故*y*的二进制编码的高*i*位为*a*1*a*2*…ai*-10**。**显然，*b*1*b*2*…bi*-1*bi*> *c*1*c*2*…ci*-1*ci*成立，即*x*的二进制编码的高*i*位比*y*大，因此，*x* > *y*成立。

综合上述充分性和必要性证明可知，式3-1成立。**证毕**。

式3-2充分性证明：假设*x* ≤ *y*时*E1*(*x*)∩*E0*(*y*)=不成立，即有*E1*(*x*)∩*E0*(*y*)≠成立。此时，根据式3-1的结论可知*x* > *y*成立，显然与假设*x* ≤ *y*矛盾，因此假设不成立，则在*x* ≤ *y*时必定有*E1*(*x*)∩*E0*(*y*)=成立。

式3-2必要性证明：假设*E1*(*x*)∩*E0*(*y*)=不时*x* ≤ *y*不成立，即有*x* > *y*成立。同理，根据式3-1的结论可知*E1*(*x*)∩*E0*(*y*)≠成立，显然与假设矛盾，因此假设不成立，则在*E1*(*x*)∩*E0*(*y*)=时必定有*x* ≤ *y*成立。

结合式3-2充分性和必要性证明可知，式3-2也成立。**证毕。**

定理3.1是利用0-1编码进行数值比较的理论基础。由定理3.1可知，数值*x*和*y*的线性关系比较问题，可以转化为判断0编码集合与1编码集合之间是否存在交集的问题。显然，如果将0编码集合和1编码集合中的元素都转换为具体数值，则可以简化集合相交的计算过程，下面我们给出针对0-1编码的数值化方法基本要求和相关实例。

### 3.1.3 0-1编码的数值化

为了保持0-1编码的数值比较特性，对于0-1编码的数值化方法必须满足以下基本要求：若将任意0-1编码二进制序列*p*的数值化方法记为*N*，得到的数值化0-1编码记为*N*(*p*)，则对于任意两个0-1编码二进制序列*p*1和*p*2，该数值化方法应满足：

1. *p*1=*p*2*N*(*p*1)=*N*(*p*2)
2. *p*1≠*p*2*N*(*p*1)≠*N*(*p*2)

**定理3.2**：对于都包含*w*个二进制位的数值*x*和*y*，设有符合要求的数值化方法*N*，与定理3.1类似，*N*(*E1*(*x*))∩*N*(*E0*(*y*))≠是*x* > *y*成立的充要条件，而*N*(*E1*(*x*))∩*N*(*E0*(*y*))=则是*x* ≤ *y*成立的充要条件。

**证明**：由满足要求的数值化方法*N*可知，*N*(*E1*(*x*))∩*N*(*E0*(*y*))≠等价于*E1*(*x*)∩*E0*(*y*)≠，而由式3-1可知，*x* > *y*的充要条件即为*E1*(*x*)∩*E0*(*y*)≠，因此*N*(*E1*(*x*))∩*N*(*E0*(*y*))≠也是*x* > *y*的充要条件，同理，*N*(*E1*(*x*))∩*N*(*E0*(*y*))=则是*x* ≤ *y*成立的充要条件。**证毕**。

针对0-1编码的数值化方法并不唯一，我们给出一种简单的数值化方法*N*：对于数值*x*的任一0-1编码序列*p*=*b*1*b*2…*bi**E0*(*x*)∪*E1*(*x*)，则*p*数值化后得到的二进制数为*N*(*p*)=1*b*1*b*2…*bi*，即在二进制序列*b*1*b*2…*bi*的最高位之前增加“1”。容

最后，我们结合具体示例，对基于0-1编码的数值比较策略做进一步说明。设数值8和11包含4个二进制位为1000和1011，则8和11对应的0-1编码即其数值化过程如图3.1所示。

由图3.1可知，*N*(*E0*(8))∩*N*(*E1*(11))不为空，且*E0*(8)∩*E1*(11)也不为空，由定理3.1和3.2的结论都可以得到*xy*成立。



图3.1 0-1编码及其数值化过程

根据定理3.1和3.2给出的利用0-1编码进行数值比较的方法可知，数值的大小比较问题，可以转化为判断数值的0-1编码集合是否相交的计算问题。

## 3.2 HMAC算法

消息认证是多媒体信息安全技术的一个重要方面，主要解决数据在通信和存储过程的完整性问题，以确保信息不受非法攻击和篡改。由于认证技术提供了通信双方身份和通信内容、过程的可信度保证，从而被广泛应用于以多媒体信息为主要交流方式的网络业务系统中，如电子商务、电子政务等。而在上述系统中，由于信息往来比较频繁，而且信息权限分级比较细，因此如何保证信息的安全性、完整性和可靠性，是认证系统必须要解决的问题。目前，消息认证技术有很多，如 HMAC、数字签名、易损水印、生物认证等。

HMAC是密钥相关的Hash消息认证码(Hash-based Message Authentication Code，HMAC)，HMAC算法利用哈希算法，以一个密钥和一个消息为输入，生成一个消息摘要作为输出。根据HMAC算法所采用哈希算法的不同，HMAC又可以进一步分为基于MD5散列函数的HMAC-MD5、基于SHA-1散列函数的HMAC-SHA1、基于SHA-256散列函数的HMAC-SHA256等。本毕设所提改进的面向隐私保护的最值查询方案结合了0-1编码方案和HMAC消息认证技术的优点，对0-1编码集合中的元素计算其HAMC消息认证码并发送给存储节点，保证了存储节点在不知道0-1编码集合元素的情况下也能按照0-1编码的性质，判断两个数之间的大小关系。

### 3.2.1 散列函数

散列函数又称作哈希函数，英文名称为Hash，散列函数的作用就是把任意长度的输入(又叫做预映射)，通过[散列算法](http://baike.baidu.com/view/31871.htm)，变换成固定长度的输出，该输出就是[散列值](http://baike.baidu.com/view/4179833.htm)。这种转换是一种[压缩映射](http://baike.baidu.com/view/8766100.htm)，也就是说[散列值](http://baike.baidu.com/view/4179833.htm)的空间通常远小于输入的空间，不同的输入可能会散列成相同的输出，而不可能从散列值来唯一的确定输入值，简单的说就是一种将任意长度的消息压缩到某一固定长度的[消息摘要](http://baike.baidu.com/view/2396437.htm)的[函数](http://baike.baidu.com/view/15061.htm)，目前常用的哈希函数有MD5和SHA-1。

所有[散列函数](http://baike.baidu.com/view/131153.htm)都有如下一个基本特性：如果两个散列值是不相同的(根据同一函数)，那么这两个散列值的原始输入也是不相同的。这个特性是[散列函数](http://baike.baidu.com/view/131153.htm)具有确定性的结果。但另一方面，[散列函数](http://baike.baidu.com/view/131153.htm)的输入和输出不是一一对应的，如果两个散列值相同，两个输入值很可能是相同的，但并不能绝对肯定二者一定相等。一个设计优秀的[加密散列函数](http://baike.baidu.com/view/6271267.htm)是一个“单向”操作，即对于给定的散列值，没有实用的方法可以计算出一个原始输入，也就是说很难伪造原始输入。以加密散列为目的设计的函数，如MD5，被广泛地用作检验[散列函数](http://baike.baidu.com/view/131153.htm)。

目前，Hash主要用于信息安全领域中加密算法，它把一些不同长度的信息转化成杂乱的128位的编码里，叫做Hash值。也可以说，Hash就是找到一种数据内容和数据存放地址之间的映射关系。Hash算法在信息安全方面的应用主要体现在以下的3个方面：

1. 文件校验

我们比较熟悉的校验算法有奇偶校验和CRC校验，这两种校验并没有抗数据篡改的能力，它们一定程度上能检测并纠正数据传输中的信道误码，但却不能防止对数据的恶意破坏。

MD5 Hash算法的"数字指纹"特性，使它成为目前应用最广泛的一种文件完整性校验和(Checksum)算法，不少Unix系统有提供计算md5 checksum的[命令](http://baike.baidu.com/view/446604.htm)。

1. [数字签名](http://baike.baidu.com/view/7626.htm)

Hash算法也是现代密码体系中的一个重要组成部分。由于非对称算法的[运算速度](http://baike.baidu.com/view/148612.htm)较慢，所以在数字签名协议中，[单向散列函数](http://baike.baidu.com/view/6321298.htm)扮演了一个重要的角色。对Hash值，又称"数字摘要"进行数字签名，在统计上可以认为与对文件本身进行数字签名是等效的。而且这样的协议还有其他的优点。

(3) 鉴权协议

鉴权协议又被称作"挑战--认证模式，在传输信道是可被侦听，但不可被篡改的情况下，通过对信息进行加密并计算其散列值，保证信息的隐私和完整性，是一种简单而安全的方法。

### 3.2.2 HMAC算法的定义

HMAC是一种执行“校验和”的算法，它通过对数据进行“求和”来检查数据是否被更改了。在发送数据之前，HMAC算法对数据块和双方约定的公钥进行散列函数计算，生成称为摘要的东西，附加在待发送的数据块中。当数据和摘要到达其目的地时，就使用 HMAC来生成另一个校验和。如果两个数字相匹配，那么数据未被做任何篡改。否则就意味着数据在传输或存储过程中，被某些居心叵测的人作了手脚。简单地讲，校验和类似于指纹，它根据消息中的数据生成一个唯一的摘要。正如指纹一样，如果消息块中有一个字节改变了，那么哈希算法将会在另一端检测到这一改变，因为从统计上可以认为每个散列值是独一无二的。

定义HMAC需要一个用来加密的散列函数(表示为*H*，可以是MD5或者SHA-1)和一个密钥*K*。我们用B来表示数据块的字节数。(以上所提到的散列函数的分割数据块字长B=64)，用L来表示散列函数的输出数据字节数(MD5中L=16,SHA-1中L=20)。鉴别密钥的长度可以是小于等于数据块字长的任何正整数值。应用程序中使用的密钥长度若是比B大，则首先用使用散列函数H作用于它，然后用H输出的L长度字符串作为在HMAC中实际使用的密钥。一般情况下，推荐的最小密钥*K*长度是L个字节。

我们定义两个固定且不同的字符串*ipad*，*opad*(‘i','o'标志内部与外部)：*ipad* = 字节0x36重复B次；*opad* = 字节0x5C重复B次。则比特流*text*的HMAC计算结果可以表示为：

*HAMC*(*K*, *text*) = *H*( *K* XOR *opad*, *H*(*K* XOR *ipad*, *text*))

即为以下运算步骤，运算过程如下：

1. 在密钥*K*后面添加0来创建一个字长为B的字符串。(例如，如果*K*的字长是20字节，B＝64字节，则*K*后会加入44个零字节0x00)。
2. 将上一步生成的B字长的字符串与*ipad*做异或运算。
3. 将数据流*text*填充至第二步的结果字符串中。
4. 用*H*作用于第三步生成的数据流。
5. 将第一步生成的B字长字符串与*opad*做异或运算。
6. 再将第四步的结果填充进第五步的结果中。
7. 用*H*作用于第六步生成的数据流，输出最终结果

### 3.2.3 HMAC安全性分析

从上文的介绍中我们可以发现，HMAC能够借助于Hash算法，对不同的输入比特流都能产生可以认为是独一无二的数字指纹。然而中国山东大学的王小云教授在2004年美国加州圣巴巴拉召开的国际密码学会议(Crypto’2004)上，做了破译破译MD5、HAVAL-128、MD4和RIPEMD算法的报告，并且在2005年王小云、尹依群、于红波等三人的论文，证明了SHA-1在理论上也被破解。王小云教授的研究成果作为密码学领域的重大发现使原本固若金汤的世界通行密码标准MD5和SHA-1的堡垒轰然倒地，引发了密码学界关于MD5和SHA-1产品是否还能够使用的大辩论。那么，使用MD5和SHA-1的HMAC算法是不是也已经不安全，本节将对HMAC的安全性进行简单分析。

我们所讨论的HMAC算法虽然使用了Hash算法，但同时也引入了秘钥加密机制，并且在本毕设提出的隐私保护查询方案中，采用的是0-1编码HAMC结合的方式实现秘密比较，使得HMAC在本毕设提出的隐私保护查询方案中的安全性已经不完全依赖于所使用的Hash算法，而由以下几点保证：

(1) 在攻击者入侵了存储节点的情况下，即使攻击者获得了HMAC加密秘钥*K*，也不能得到比特流*text*的值。这是因为王教授的研究成果是在已知输出时，可以较容易地伪造出一个输入，使输入经过Hash后与已知的输出一致，而不是计算出原本的输入，从而保证了数据的0-1编码元素和数据本身的隐私。

(2) 王教授的研究成果在于构造一个伪输入，而输出结果却是相同的，若入侵存储节点的攻击者没有改变0-1编码元素的HMAC运算结果，那么存储节点在执行查询后，查询结果相对于未入侵前是不会改变的。

HMAC算法在最值查询协议中还可以起到保护数据完整性的作用，而本毕设关注的是在最值查询过程中数据的隐私性，所以这里对HMAC算法在数据完整性方面的应用就不做详细介绍。

## 3.3 对称加密技术

随着计算机网络技术的迅速发展，网上数据通信越来越频繁，为了保证数据在传感器网络上传输时不被窃取或篡改，就有必要对数据进行加密，以保证数据的安全传输。而所谓的加密，就是用基于数学方法的程序和保密的密钥对信息进行编码，把计算机数据变成一堆杂乱无章难以理解的字符串，只允许合法的接收者恢复数据的本来面目，而对于非法窃取者，即使得到了密文，也无法辨认原始数据。

加密体制按照加密密钥和解密密钥之间关系可以分为对称密码体制和公钥密码体制，对称密码主要包括分组密码和流密码，具有运行速度快,存储量小和易于软硬件实现等优点。

本毕设改进的面向隐私保护的最值查询协议是基于两层传感器网络的，而两层传感器网络的优点在于把复杂的查询处理过程放在具有充足能源和强大计算能力的存储节点，使得在降低大量布置的感知节点的成本同时，进一步地提高了整个传感器网络的数据处理能力，并且使得传感器网络的负载更加均衡。然而为了保护数据的隐私，感知节点需要借用计算机网络通信技术中的加密体制来实现数据加密，同时为了减少感知节点的工作负担，采用的加密体制不能过于复杂。故而，为了尽量减小感知节点的工作量同时又提高其处理速度，本毕设采用了对计算能力要求相对较低而且运行速度相对较快的DES对称加密算法，作为感知节点的数据加密算法。

### 3.3.1 DES基本概念

DES算法为对称密码体制中的一种分组密码。在加解密过程中使用同一种算法，一般情况下，发送方先生成随机对称密钥对数据进行加密形成复杂的加密密文后，连同密钥一同发送给接收方，接收方收到密文后，使用加密时使用的密钥及相同算法的逆算法对密文进行解密，最终得到明文。在本毕设所改进的的最值查询协议中，由于加密算法的使用环境是两层传感器网络，所以每个感知节点在随机撒放到环境中前，就与Sink节点协商好本节点的对称秘钥。

DES算法的一个主要特点是不需要同步，因而在分组交换网中有着广泛的用途。目前在国内，随着三金工程尤其是金卡工程的启动，DES算法在POS、ATM、磁卡及智能卡(IC 卡)、加油站、高速公路收费站等领域被广泛应用，以此来实现关键数据的保密，如信用卡持卡人的PIN的加密传输，IC卡与POS间的双向认证、金融交易数据包的MAC校验等，均用到DES算法。

### 3.3.2 DES算法工作原理

DES采用算法混乱和扩散的组合技术(先替代后置换)，先对整个的明文进行分组。每一个组长为64位，采用16轮迭代，对每一个64位二进制数据进行加密处理，产生一组64位密文数据，并将各组密文串接起来，即得出整个的密文，达到对数据进行加密的目的。

算法的加密流程为：

1. 输入64位的明文数据块。
2. 初始置换。把输入的64位数据块按位重新组合，并把输出分为两部分左明文和右明文，每部分各长32位，其置换规则如图3.2所示。

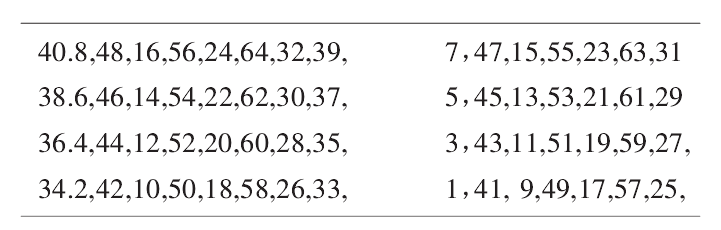


图3.2 初始置换和逆置换

亦即将输入的第58位换到第1位，第50位换到第2位，...，依次类推，最后一位是原来的第7位，L0，R0则是换位输出后的两部分，L0是输出的左32位，R0是右32位。如假设置换前输入值为D1D2D3...D64，则经过初始置换后的结果为：L0=D58D50...D8；R0=D57D49 ... D7。

1. 迭代。在有密钥控制下执行DES的16轮迭代，并把得到的结果交换左、右各32 bit的位置。
2. 逆置换(初始置换的逆运算)。将经过16次迭代运算后得到的L16和R16作为输入进行逆置换，即得到1个64位明文加密后的密文。

DES加密流程如图3.3所示。

其中，“扩展置换”是将右明文从32bit扩展到48bit；密钥是将最初64 bit密钥通过放弃每个第8位得到56bit密钥，每轮迭代从这个56bit密钥产生不同的48bit子密钥，并与48bit的右明文进行异或运算，将结果传递到下一步做S盒置换。S盒再将48bit的输入压缩为32bit的输出，由P盒进行32bit到32bit的简单换位，将P盒置换的结果与左明文进行异或运算便得到新的右明文，并通过交换将旧的右明文变为新的左明文，这样一轮迭代的任务即宣告完成。



图3.3 DES算法加密流程图

DES算法加密流程中F(Ri,Ki)的算法描述如图3.4所示。



图3.4 F(Ri,Ki)的算法描述

由于DES是对称加密算法，所以它的解密流程正好是加密流程的逆过程，这里就不再对DES解密流程进行描述。

## 3.4 本章小结

本章对隐私保护最值查询协议关键技术进行了详细的描述，包括0-1编码技术、HMAC算法和DES对称加密技术。为了保护数据隐私，感知节点需要对感知数据进行DES加密处理，同时为了给存储节点提供足够的信息以进行处理，感知节点还需要提供数据的0-1编码，但由于0-1编码具有可逆推的特性，因此在发送0-1编码前，感知节点还需要对数据的0-1编码集合做HMAC运算处理。本章对0-1编码的数值比较特性、DES的加密过程和HMAC算法的安全性都进行了详细的分析，为第四章中协议研究做理论准备。

# 第四章 改进的隐私保护最值查询协议

## 4.1 威胁模型与协议模型

两层传感器网络相比于传统传感器网络而言，具有拓扑结构简单和存储节点资源充裕等特点。由于采用了分层结构的思想，两层传感器网络具有链路质量稳定、路由结构简单、查询高效和负载均衡等优点。然而在存储节点发挥着日益重要作用的同时，其在数据处理过程中的突出地位又使得它成为了攻击者的首要目标。本节将对两层传感器网络所面临的威胁与问题进行详细分析，同时初步设计了改进的最值查询协议以保护传感器网络的数据隐私。

### 4.1.1 威胁模型与问题描述

在两层传感器网络中，由于存储节点不仅存储着大量的感知数据，同时还负责执行来自Sink节点的查询请求，这就使得存储节点往往成为恶意攻击的首要目标。在查询处理过程中，如果不对感知数据进行任何处理，而是直接以明文形式参与查询处理过程，那么当存储节点被俘获时，攻击者将能够获取所有存储于其中的感知数据。尽管感知节点同样可能被俘获，然而，由于单个感知节点产生的数据相对与整个网络而言非常少，即使存在少量感知节点被俘获，对整个网络的影响也不大[28]。因此，我们重点关注存储节点被俘获的隐私保护措施。

与现有工作类似，本毕设同样假设Sink节点和感知节点可信，而存储节点不可信，且存储节点有试图窥探其他节点数据的企图，但仍然能够遵循查询处理协议执行，即符合Honest-but-curious威胁模型[29]。

在上述假设条件下，实现具有隐私保护能力的最值查询，就必须确保查询处理过程满足：对于网络中的任一感知节点的感知数据以及查询结果，只有Sink节点可以获取其明文数值，而存储节点无法获取。

实现上述要求的关键在于：如何使存储节点在无需感知数据明文数值参与的情况下，相互之间比较大小关系。改进具有隐私保护能力的两层传感器网络最值查询。

### 4.1.2 协议模型

由第三章介绍的0-1编码的原理及相关定理可知，我们可以将数值间的大小比较转化为0-1编码集合的关系比较。然而根据0-1编码的相关定理，攻击者在攻陷存储节点的情况下，可以通过任意数的0-1编码集合推知该数的原值，所以我们还需要借助Hash消息认证机制，利用其单向性和抗碰撞的特性，对0-1编码集合的元素进行Hash消息认证计算以实现秘密比较。为便于Hash消息认证的计算，我们需要在进行HMAC计算之前对0-1编码的二进制序列进行数值化计算。本毕设重点在于降低感知节点能耗，我们采取随机选取每个感知数据的0-1编码的方法减少数据传输数量。由于最大值和最小值方法类似，在本协议中只研究最大值。

综上，我们可以初步建立最值查询协议模型OMQP，如图4.1所示。



图4.1 最值查询协议模型

(1) 感知节点执行数据上传协议，对收集到的数据进行加密，并对数据进行0-1编码计算、数值化计算和HMAC计算，然后将加密的数据和经过HMAC处理的0-1编码上传给存储节点；

(2) Sink节点将查询处理协议发送给存储节点；

(3) 存储节点执行查询处理协议，对感知数据进行比较后，判断感知数据的大小，最后将较大的感知数据的密文集发送Sink节点；

(4) Sink节点对存储节点上传的数据进行解密，比较大小并获得最大值，最终将数据返回给用户。

我们将在后续章节对OMQP的具体实现细节进行详细的介绍，并最终形成完整的隐私保护最值查询协议。

## 4.2 数据上传协议

在感知节点数据上传阶段，感知节点在每一个时间槽内，获取感知数据，并对感知数据进行加密和编码处理，然后将处理后的数据上传至其所在单元内的存储节点。为了便于表示，本毕设把数值*x*的1编码记为*E1*(*x*)，0编码集合记为*E0*(*x*)；对1编码集合或0编码集合的数值化计算表示为*N*(*E1*(*x*))或*N*(*E0*(*x*))；对任意输入*x*用秘钥*g*进行HMAC计算过程用*HMACg*(*x*)表示；用符号*M*表示存储节点。

数据上传协议具体过程如协议4.1所示：

|  |
| --- |
| **协议4.1：数据上传协议** |
| 对于单元*C*=(*M*, {*s*1, *s*2, …, *sn*})内的任意感知节点*si*，设*si*在*t*时间槽内采集到的*N*个感知数据为*Di*={*di1*,*di2*, …,*diN* }，计算其最大值，记为*di*=max(*Di*)。在*t*结束之前，*si*依次执行如下操作：   1. *si*利用与Sink节点共享的密钥*ki*，对*di*进行DES加密计算，得到密文数据*Enc*(*di*, *ki*)； 2. 对于*di*，用全网共享秘钥*g*进行HMAC数值化0-1编码处理，得到*HMACg*(*N*(*E0*(*di*)))和*HMACg*(*N*(*E1*(*di*)))。 3. 将如下数据消息发送给*M*，其中*id*(*si*)表示*si*的ID信息，*rnd*(*X*, *Y*)表示随机取*X*和*Y*中任一元素。   *si*→*M*：< *id*(*si*), *t*, *Enc*(*di*, *ki*), *rnd*{ *HMACg*(*N*(*E0*(*di*))), *HMACg*(*N*(*E1*(*di*)))}>  当存储节点*M*接收到*si*上传的数据时，就立即将这些数据存储起来。 |

为了描述的简洁性，本毕设分别称*HMACg*(*N*(*E0*(*di*)))和*HMACg*(*N*(*E1*(*di*)))为数据*x*的0编码和1编码的比较因子，而*rnd*{ *HMACg*(*N*(*E0*(*di*))), *HMACg*(*N*(*E1*(*di*)))}则称为随机比较因子。

由协议4.1可知，感知节点产生的所有感知数据都将进行加密和Hash消息认证计算，然后在发送并存储于存储节点*M*，这就使得*M*无法获取任何感知节点的感知数据；在数据上传过程中，分别有*n*个密文数据和比较因子发送至*M*。此外，由于HMAC机制的单向性和抗冲突性，经过HMAC数值化处理后的数据依然可以进行大小比较，即有如下定理4.1成立。

**定理4.1：**对于数值*x*和*y*，有如下推导关系成立：

1. *HMACg*(*N*(*E1*(*x*)))∩*HMACg*(*N*(*E0*(*y*)))≠*x*>*y*
2. *HMACg*(*N*(*E0*(*x*)))=*HMACg*(*N*(*E0*(*y*)))*HMACg*(*N*(*E1*(*x*))) =*HMACg*(*N*(*E1*(*y*)))*x*=y
3. *HMACg*(*N*(*E0*(*x*)))≠*HMACg*(*N*(*E0*(*y*))*HMACg*(*N*(*E1*(*x*)))≠

*HMACg*(*N*(*E1*(*y*)))*HMACg*(*N*(*E1*(*x*)))∩*HMACg*(*N*(*E0*(*y*))) =*x*<*y*

由定理3.1以及HMAC机制的抗碰撞性，容易证明定理4.1成立。

## 4.3 查询处理协议

在查询处理阶段, Sink节点将查询指令发送给存储节点；存储节点利用定理4.1，比较自身存储的密文数据之间的大小关系。由于同种编码不能比较大小，可能存在*M*中几个同种编码都比另一种编码大，不能确定最大值，我们将这些密文数据返回给Sink节点；Sink节点解密存储节点返回的密文数据，进一步比较大小，得到最终的查询结果。

查询处理协议具体内容如协议4.2所示：

|  |
| --- |
| **协议4.2：**查询处理协议 |
| **Sink节点：**  (1) 设用户的查询指令*Q* =(Ψ, *t*)，其中Ψ表示查询覆盖的感知节点集合，*t*为查询的时间槽。Sink节点将查询指令发送给存储节点，并等待存储节点的查询反馈；  (2) 当Sink节点接收到*M*发来的数据消息后，利用与感知节点共享的密钥解密返回的密文数据，进一步比较大小，获得最终的最大值。  **存储节点*M***：  当存储节点接收到查询指令*Q*=(Ψ, *t*)后，*M*依次获取每一个被查询的感知节点在*t*内上传的密文数据和对应的HMAC数值化0-1编码，利用定理4.1，并根据最值密文候选集生成算法（见算法4.1），比较密文数据之间的大小关系，从而获得密文查询结果*RC*。然后，*M*构造如下消息发送至Sink节点：  *M*→Sink：< *RC* > |

由协议4.2可知，*M*上传Sink节点的密文查询结果*RC*不为空(必然有最值)，且最多有*n*个密文数据，即0<|*RC*| ≤*n*。

|  |
| --- |
| **算法4.1：最值密文候选集生成算法** |
| 存储节点*M*创建0编码集合*R*0和1编码集合*R*1，*R*0和*R*1初始状态为空集。*M*按顺序提取*HMACg*(*N*(*E*(*di*)))，并判断其编码种类。将*HMACg*(*N*(*E0*(*di*)))放*R*0，将*HMACg*(*N*(*E1*(*di*)))放入*R*1，当*R*0和*R*1都不为空集时，对余下的*HMACg*(*N*(*E*(*di*)))进行如下操作：  (1) 判断*HMACg*(*N*(*E*(*di*)))的编码种类，如果是0编码，则根据定理4.1与*R*1中的所有元素比较大小，如果*HMACg*(*N*(*E0*(*di*)))比*R*1中的所有元素大，则将*HMACg*(*N*(*E0*(*di*)))放入*R*0 ；如果*HMACg*(*N*(*E0*(*di*)))比*R*1中的部分元素大，则将小于*HMACg*(*N*(*E0*(*di*)))的元素和*HMACg*(*N*(*E0*(*di*)))一起删除；如果*HMACg*(*N*(*E0*(*di*)))比*R*1中的所有元素小，则只删除*HMACg*(*N*(*E0*(*di*)))。*HMACg*(*N*(*E1*(*di*)))编码同理。  (2) 当全部*HMACg*(*N*(*E*(*di*)))分配完毕后，根据定理4.1，将*R*0中的每个元素分别与*R*1中所有元素比较大小。如果*HMACg*(*N*(*E0*(*di*))) 比*R*1中所有元素大，则将其对应的<*id*(*si*), *Enc*(*di*, *ki*)>加入*RC*；如果*HMACg*(*N*(*E0*(*di*))) 比*R*1中部分元素大，则将小于*HMACg*(*N*(*E0*(*di*)))的元素和*HMACg*(*N*(*E0*(*di*)))一起删除；如果*HMACg*(*N*(*E0*(*di*)))比*R*1中的所有元素小，则只删除*HMACg*(*N*(*E0*(*di*)))。  (3) 当*R*0中全部*HMACg*(*N*(*E0*(*di*)))与*R*1中所有元素比较完毕后，如果*RC*不为空集，则*RC*为查询结果；如果*RC* 为空集，则将*R*1中所有剩余元素对应的<*id*(*si*), *Enc*(*di*, *ki*)>加入*RC*。 |

## 4.4 协议分析

我们在4.1节、4.2节和4.3节中提出并完善了改进的两层传感器网络隐私保护最值查询协议，并将其建模为OMQP。本节将对OMQP的数据隐私保护强度及能量消耗表现进行科学的分析，以检验其现实可行性，为OMQP的应用和前景提供理论依据。

### 4.4.1 隐私安全性分析

我们分别从感知数据、查询结果两个方面分析OMQP的隐私安全性。

(1) 感知数据的隐私安全性

在本毕设基于可信Sink节点的前提下，对于任意感知节点产生的感知数据而言，仅当查询处理方法能够确保存储节点无法获取其明文数值时，感知数据才是隐私安全的：首先，在查询处理过程中，感知节点在传输感知数据时，需首先对感知数据进行加密处理(密钥仅与Sink节点共享)。因此，在没有密钥的情况下获取感知数据的复杂度与破解加密算法相同，从而确保任意传输路径中的其它节点都无法获取其明文数值。其次，本毕设利用HMAC机制，对感知节点发送给存储节点的数值化0-1编码数据进行HMAC处理，使得存储节点在无需感知数据明文参与的情况下，即可实现各感知数据的大小比较，从而确定出包含查询结果的候选密文数据集；而HMAC机制的单向性，使得存储节点无法利用获得的HMAC数据反向逆推其对应的感知数据。可见，本毕设改进的最值查询协议能够确保感知数据的隐私安全性。

(2) 查询结果的隐私安全性

查询结果的计算由存储节点和Sink节点协作完成，存储节点利用定理4.1比较各感知节点上传的各感知数据比较因子的大小，进而计算出包含查询结果的候选密文数据集，并发送给Sink节点，最终由Sink节点解密候选密文数据集中的密文数据，进一步比较，从而计算出最终的查询结果。可见，存储节点的查询处理执行过程，只涉及到HMAC数据和密文数据，由于没有密钥信息，存储节点无法获得任何感知数据的明文数值，因此也就无法获得任何查询结果数据。可见，OMQP同样能够确保查询结果的隐私安全性。

由于本毕设提出的最值查询协议和文献[7]中的SafeQ都采用了对称加密和HMAC机制，因此，这两种方案在隐私安全性上具有同等的保护能力。

### 4.4.2 能耗分析

在两层传感器网络中，由于存储节点具有充足的能量储备，而感知节点能量资源受限，整个网络的生命周期主要受感知节点的能耗影响。因此，我们主要从感知节点能耗角度，给出协议的能耗分析。

通常情况，感知节点的能耗主要由发送、接受数据消息产生，即通信能耗，记为*EOMQP*。

假设网络中存在*N*个感知节点，每个感知节点单位时间槽内产生*n*个感知数据，每个感知数据包含*w*个二进制位，单位密文数据和HMAC数据的长度分别为*lc*和*lh*，接收和发送单位数据的平均能耗分别为*er*和*es*，感知节点到存储节点的平均路径长度(跳数)为*L*，感知节点ID和时间槽数据长度分别为*lid*和*lt*。

由协议4.1可知，任一感知节点*si*将上传两类数据给存储节点：一是*n*个密文数据；二是*n*个随机选择的比较因子，假设*si*产生的第*j*个感知数据对应的随机比较因子包含*ρi, j*个HMAC数据，则通信能耗*EOMQP*为：

 (式4-1)

假设HMAC机制随机分布，再由3.1.2节性质3.1(3)可知，任一包含*w*个二进制位的数据*x*的0编码和1编码数据之和为*w*，因此*x*的随机比较因子*rnd*{*HMACg*(*N*(*E0*(*x*))), *HMACg*(*N*(*E1*(*x*)))}平均包含*w*/2个HMAC数据，因此*EOMQP*可近似为：

 (式4-2)

与EMQP相比，由于EMQP同样采用对称加密和HMAC机制，而EMQP中需上传每个感知数据的所有比较因子，即HMAC数值化0编码和1编码数据，将使得每个感知节点产生*w*个HMAC数据(*w*为感知数据的二进制位数)，因此OMQP具有更高的能耗使用效率。我们将在第下一节的实验部分给OMQP和EMQP的感知节点能耗对比和分析。

## 4.5 实验分析

为了对协议的性能进行比较和分析，本毕设在文献[30]的仿真器上实现了EMQP和OMQP。通过模拟仿真，我们给出EMQP和本毕设的感知节点能耗的实验结果及分析，并给出本毕设中的通信和计算能耗实验结果及分析。本毕设的实验环境为Pentium E5700(双核3.0GHz)CPU，3G内存；软件环境为Windows XP操作系统，Eclipse和Matlab；实验数据集通过随机产生。

我们采用文献[31]中的能耗计算方法：无线通信电路发送和接收1byte的能量消耗公式为*es = α + γdm*和*eγ = β*，其中，*α*为通信发送电路消耗的能量，*γ*为传输放大器消耗的能量，*d*为传输距离，*m*为路径损失因子，*β*为通信接收电路消耗的能量。参数设置与文献[30]相同：*γ*=10pJ/bit/m2，*α*=45nJ/bit，*β*=135nJ/bit，*m*=2。其它参数设置如下表所示。

表4.1 实验参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 参数值 | 参数名 | 参数值 |
| 网络覆盖区域 | 100x100 m2 | 感知数据长度(*w*) | 10b |
| 节点通信半径 | 10 m | 密文数据长度 | 16B |
| 感知节点分布 | 随机分布 | HMAC数据长度 | 16B |
| 实验次数(网络ID) | 20 | 时间槽和感知节点ID长度 | 4B |
| 感知节点数量(*n*) | 480 | 单位时间槽产生数据量(*N*) | 50 |

在每一组实验中，我们通过在网络覆盖区域中随机分布感知节点，生成20组不同拓扑结构的网络(用不同的网络ID标识)，进而通过计算20组网络的平均能耗值确定最值查询的总能耗。具体实验结果及分析如下：

(1)在实验设置的初始参数条件下，随机生成的20组网络的能耗实验结果如图4.2所示。



图4.2 不同网络拓扑下的能量消耗

由图4.2可知，在不同拓扑的网络中，OMQP和EMQP的总能耗均变化不大，且分布都较为均匀，但EMQP显著高于OMQP，在实验设置的条件下，OMQP的总能耗平均约为EMQP的56.5%，其主要原因在于，OMQP中的感知节点需上传的HMAC数据量显著少于EMQP。

(2) 以感知节点数量*n*为自变量，其他参数保持初始设置不变，得到图4.3所示的实验结果。



图4.3 以*n*为自变量时的能量消耗

由图4.3可知，当*n*增大时，OMQP和EMQP的总能耗均随之增大，且EMQP的增长幅度明显高于OMQP，在实验设置的条件下，OMQP的总能耗平均约为EMQP的56.5%，其原因与(1)类似。

(3) 以*w*为自变量，其他参数保持初始设置不变，得到如图4.4所示的实验结果。

6



图4.4 以*w*为自变量时的能量消耗

由图4.4可知，随着*w*的增大，OMQP和EMQP的总能耗均随之增长，但OMQP增长缓慢，而EMQP的增长较快，在实验设置的条件下，OMQP的总能耗平均约为EMQP的54.29%，其原因与(1)类似。

综合上述实验结果及分析可知：与EMQP相比，本毕设提出的OMQP方法的感知节点总能耗显著较低，在实验设置的条件下，平均约为前者的55%。

## 4.6 本章小结

本章阐述了隐私保护最值查询OMQP的实现思路，包括威胁模型描述、问题描述与协议模型，对第三章基于0-1编码的数值比较原理进行拓展，得到HMAC加密之后数据的安全比较原理。详细介绍了数据上传协议和查询处理协议，并对OMQP的隐私安全性和能耗表现进行了分析。在模拟系统中模拟运行OMQP并分析其在模拟环境下的能耗表现，同时与EMQP协议进行了通信能耗比较，体现本毕设较现有技术的改进和优化之处。

# 第五章 OMQP模拟系统的设计与实现

## 5.1 模拟系统设计框架

本程序使用Microsoft Visual Studio2010作为开发工具，使用C#作为软件开发语言，设计并实现了OMQP模拟系统。VS2010集成了Windows窗体应用程序开发工具，使我们能够把主要精力放在OMQP模拟系统的数据结构和函数设计上，下面我们将介绍模拟系统的界面设计、模拟系统设计背景和数据结构设计。

### 5.1.1 模拟系统界面设计

由于VS2010集成了强大的窗体程序设计控件，所以我们可以设计出如图5.1所示的程序界面。

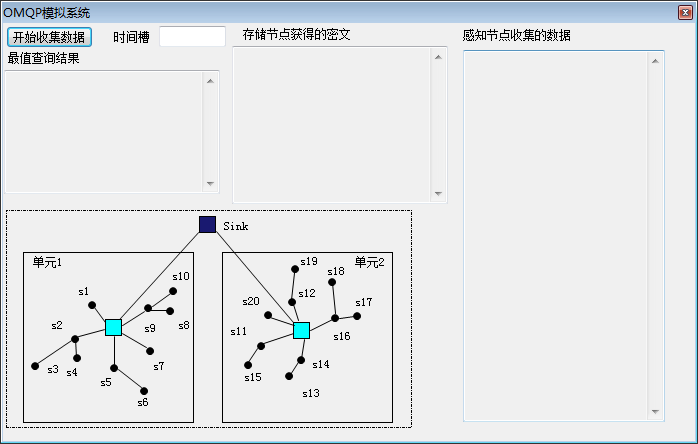


图5.1 模拟系统界面设计图

由于本毕设研究的是一种改进的两层传感器网络隐私保护最值查询方法，所以本模拟系统不考虑传感器网络路由动态建立过程，而以一个静态路由作为两层传感器网络工作路由。

在图5.1中我们可以看到最值查询结果、存储节点获得的密文和感知节点收集的数据，三个只读文本框，其中感知节点收集的数据文本框中显示内容是感知节点收集到的数据；存储节点获得的密文文本框主要显示的是存储节点收到来自感知节点的信息；最值查询结果文本框显示的是在数据收集结束后，模拟系统根据时间槽中用户输入的查询时间进行查询得到的查询结果。

### 5.1.2 模拟系统设计背景和数据结构设计

本模拟系统模拟现实环境中两层传感器网络实现OMQP过程，共模拟设置了一个Sink节点、两个存储节点和20个感知节点，每个存储节点负责一个单元中10个感知节点的数据存储和查询处理。感知节点每隔500ms收集一次整数数据，以1s为一个时间槽，共收集10个时间槽，在每个时间槽收集数据后对数据进行相应的处理并将处理结果发送给存储节点。

在实际情况下，传感器网络各节点间传输数据需要通过无线网络实现，而在本设计的模拟系统中，我们采用共享数组的形式实现节点间通信。由于在两层传感器网络中，各个节点的功能可以被划分为存储功能和计算功能，所以本设计用不同的数组和函数表示各节点以及各节点。本设计模拟系统的主要全局变量如下。

表5.1主要全局变量

|  |  |
| --- | --- |
| 变量名 | 功能 |
| *global\_time* | 统计一共收集了几个时间槽的数据 |
| *maxvalue* | 表示收集数据的最大值 |
| *aTimer* | 定时器*aTimer*用于500毫秒定时，每隔500ms将会触发一个事件 |
| *ran* | 随机对象*ran*用于随机产生感知数据 |
| *keyofhmac* | 数组*keyofhmac*用于存放全网共享的HMAC秘钥 |
| *keyofdes* | 数组*keyofdes*用于存放各感知节点与Sink节点共享的DES秘钥 |
| *sensor\_data* | 三维数组*sensor\_data*用于存放各感知节点收集的数据 |
| *secret\_data* | 数组*secret\_data*存放各感知节点上经过DES加密的数据 |
| *zo\_code* | 数组*zo\_code*存放各感知数据的0编码和1编码集合中的最小集合 |
| *hmac* | 数组*hmac*用来存放以集合*zo\_code*对应元素为输入，以*keyofhmac*为秘钥的HMAC计算结果 |

## 5.2 主要函数的设计与实现

### 5.2.1 0-1编码函数实现

0-1编码函数*zo\_encode*(byte *a*, ref byte[ ][ ] *str*)的功能是对数据*a*进行0-1编码计算并将结果存放在数组*str*中，并返回0编码集合元素个数。函数首先将输入数据*a*转化为二进制形式并存储在数组*value*中，然后以数组*arry*为临时空间，根据数据的二进制形式分别进行0编码计算和1编码计算。

0编码计算和1编码计算流程如下所示：



1. 0编码



(b) 1编码

图5.2 0-1编码计算流程图

1. 实现0编码的核心代码如下：

|  |
| --- |
| for (i = 0; i < 8; i++) {  if (value[i] == 0) {  arry[0] = 1;  for (k = i + 1; k < 8; k++)  arry[k - i] = value[k];  k = 8 - i;  str[j] = new byte[k];  for (int h = 0; h < k; h++)  str[j][h] = arry[h];  j++;  }  } |

1. 实现1编码的核心代码如下：

|  |
| --- |
| for (i = 0; i < 8; i++) {  if (value[i] == 1) {  for (k = i; k < 8; k++)  arry[k - i] = value[k];  k = 8 - i;  str[j] = new byte[k];  for (int h = 0; h < k; h++)  str[j][h] = arry[h];  j++;  }  } |

### 5.2.2 定时器事件函数实现

定时器*aTimer*每隔500ms就会触发一次定时器事件*aTimedEvent*，该事件将会产生20个值在0到100之间的整数，作为模拟20个感知节点收集的感知数据，同时还会将计算该数据的0-1编码集合。同时若触发事件的次数是偶数，该事件就会认为一个时间槽即将结束，在产生随机数据之后会对本时间槽内收集到的数据进行DES加密计算、0-1编码计算及其Hash消息认证计算。

*aTimedEvent*的流程图如下所示。



图5.3 *aTimedEvent*程序流程图

关键代码：

|  |
| --- |
| sensor\_data[time, i, k] = (byte)ran.Next(100);  data[0] = sensor\_data[time, i, k];  secret\_data[time][i][0] = EncryptString(data, keyofdes[i]); //数据加密  count = zo\_encode(sensor\_data[time, i, k], ref temp\_zo); //0-1编码计算  if (count > 4){  zo\_state = false;  count = 8 - count;  }  for (int j = 1; j < count; j++){  hmac[time][i][k][j] = new byte[16];  hmac[time][i][k][j]=hmacmd5.ComputeHash(zo\_code[time][i][k][j]); //Hash消息认证码计算  } |

### 5.2.3 最值密文候选集生成算法

由于同种0-1编码不能比较大小，本算法将感知节点随机上传的编码分类，然后互相比较并删除较小的值，最终得到包含最大值的同种编码密文数据集。

具体算法流程图如下：



图5.4 最值密文候选集生成算法流程图

关键代码：

|  |
| --- |
| for (int i = 0; i < 20; i++){  output = "Sensor" + Convert.ToString(i + 1) + ":";  for (int k = 0; k < 2; k++){  in\_state = false;  bool zo\_state = Convert.ToBoolean(hmac[time][i][k][0][0]);  if (in\_state){  byte[] minwen = DecryptString(secret\_data[time][i][k], keyofdes[i]);  foreach (byte temp in minwen)  output += Convert.ToString(temp);  output += " ";  }  }  output += "\r\n";  SetText\_of\_queryresult(output);  }  complete = false;  bool state = false;  int a\_acount = a.Length;  int b\_acount = b.Length;  int count = 0;  if (a\_acount == 1 || b\_acount == 0) { //其中任何一方集合为空，交集为空  if (a\_acount == b\_acount + 1)  complete = true;  return state;  }  for(int i = 1; i < a\_acount; i++)  for(int j = 0; j < b\_acount; j++) {  if (compare(a[i],b[j]))  count++;  }  if (a\_acount == b\_acount + 1 && count == a\_acount - 1)  complete = true;  if (count != 0) state = true;  return state;  } |

### 5.2.4 其他函数的实现

本毕设使用.NET 4.0作为模拟系统开发环境，该开发环境集成了HMAC计算函数和DES加密函数，所以本模拟系统就不再独立设计这两个算法，而调用直接系统集成函数进行运算。

## 5.3 模拟系统的工作过程

本模拟系统模拟实际环境中最值查询协议的工作过程，包括数据收集过程和数据查询过程。数据收集过程如图5.4所示。

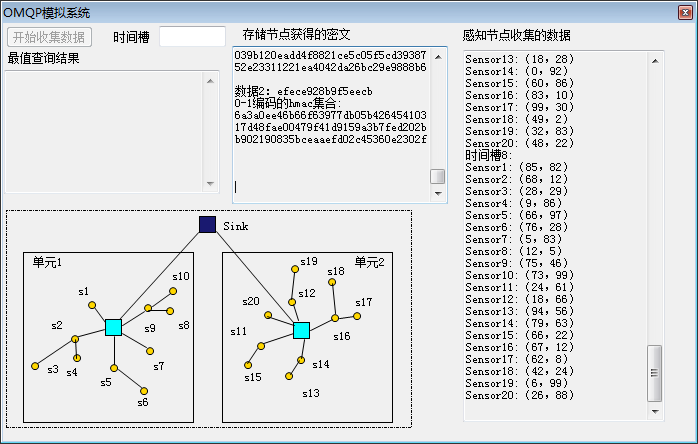


图5.4 数据收集过程

由图5.4可知，每个感知节点每500ms收集一次数据，每秒对本时间槽内收集到的数据加密并计算0-1编码Hash消息认证信息后，将相关数据发送给存储节点。显示了存储节点收到的感知数据的密文，存储节点无法获取数据明文；同时感知节点向存储节点提供了用于查询比较的随机一个0-1编码HMAC集合，和经过加密的感知数据一样，存储节点无法通过这些HMAC集合获得任何关于原始数据值的有价值的信息，从而保护了感知数据的隐私。

数据收集过程结束后，在时间槽输入查询条件，再单击开始查询按钮，查询过程及结果显示如图5.5所示。

在查询结果对话框中显示的是最值查询的结果，易知虽然存储节点无法获得感知数据的真实值，却成功计算出符合查询条件的感知数据，从而验证了本设计的可行性。

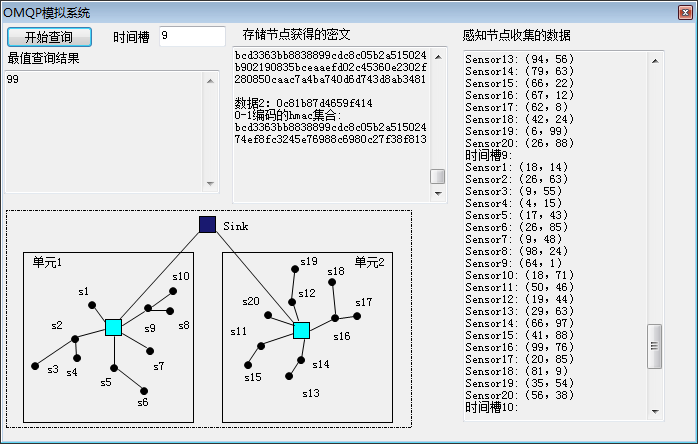


图5.5 查询结果显示

## 5.4 本章小结

本章详细介绍了隐私保护最值查询模拟系统的实现与设计过程，验证了本设计在现实环境下的可行性。从感知数据的产生过程中我们可以看到，每个感知数据向存储节点发送的0-1编码消息认证集合元素的数量等于2，即感知数据长度的一半，与EMQP相比，OMQP能够显著降低感知节点的能量消耗。

# 结束语

本毕业设计以两层传感器网络为研究背景，采用安全多方计算中的0-1编码技术和Hash身份认证编码技术，改进了具有隐私保护能力的最值查询方法，并给出针对能量消耗的模拟仿真实验；设计并实现了OMQP模拟系统验证了本毕业设计所提方案的可行性。

本次毕业设计的主要特色有：

1. 研究并分析了无线传感器网络的结构和原理，介绍了传感器网络隐私保护协议的相关工作；
2. 介绍了安全多方计算中的0-1编码技术、Hash身份认证编码技术和当前适用于两层传感器网络环境的感知数据秘密比较机制；
3. 基于秘密比较机制，并结合两层传感器网络特点，给出了改进的两层传感器网络隐私保护最值查询的数据上传协议和查询处理协议；
4. 在本毕业设计改进的两层传感器网络隐私保护最值查询技术的基础上，设计并实现了OMQP模拟系统。

由于受到各方面因素的限制，本毕设改进的隐私保护最值查询模拟协议还存在许多不足和需要改进之处。本毕设主要改进了两层传感器网络中隐私保护最值查询协议，重点减少了感知节点上传的数据量，完成了减少能耗的目标。但没有保护查询过程中感知数据的完整性的机制，因此在以后的工作中还需要对协议进行不断的完善。

# 致 谢

本设计是在戴华老师的细心指导下完成的，戴华老师虽然平日工作繁多，但不仅提前为我们仔细撰写任务书，给我们提供详细的相关资料，指导我们理解毕设内容及思想，并且坚持每周至少一次检查我们的进度，督促我们按时完成工作，及时对我们的问题进行解答和回复。在三个月的毕业设计中，戴华老师严谨细致的治学态度深深的感染了我，并教导我直面困难，克服困难，对我的人生产生了深远的影响。在此，我向戴华老师表示我衷心的感谢！

# 参考文献

[1] J. Cheng, H. Yang, S. H. Y. Wong et al. Design and implementation of cross-domain cooperative firewall [C]. Proceedings of the 15th IEEE International Conference on Network Protocols. Piscataway , NJ, USA, 2007: 284-293.

[2] A.X. Liu, F. Chen, Collaborative enforcement of firewall policies in virtual private networks [C]. Toronto, Canada, 2008.

[3] Hua Dai, Geng Yang, and Xiaolin Qin. EMQP: An Energy-Efficient Privacy-Preserving MAX/MIN Query Processing in Tiered Wireless Sensor Networks

[4] Jason L H. System architecture for Wireless sensor networks [D]. Berkeley, California, USA: University of California at Berkeley, Ph.D Thesis, 2003.

[5] Mainwaring A, Polastre J, Szewczyk R, et al. Wireless sensor networks for habitat monitoring [C]. Proceedings of WSNA, 2002, pp.88-97.

[6] Arampatzis T, Lygeros J, Manesis S. A survey of applieations of wireless sensors and wireless sensor networks [C]. IEEE International Symposium on Intelligent Control, 2005, 719-724.

[7] He w, Liu X, Nguyen H et al. PDA: Privacy-preserving data aggregation in wireless sensor networks [C]. Proceedings of the 26th IEEE International Conference on Computer Communications. Anchorage, USA, 2007: 2045-2053.

[8] Yao J B, Wen G J. protecting classification privacy data aggregation in wireless sensor networks [C]. Proceedings of the 4th International Conferences on Wireless Communication, Networking and Mobile Computing. Dalian, China, 2008: 1-5.

[9] He W B, Hoang N, Liu X et al. IPDA: An integrity-protecting private data aggregation scheme for wireless sensor networks [C]. Proceedings of the Military Communications Comference. San Diego, CA, USA, 2008: 1-7.

[10] 杨庚, 王安琪, 陈正宇等. 一种低耗能的数据融合隐私保护算法 [J]. 计算机学报, 2011,34(5): 792-800.

[11] Claude C, Einar M, Gene T. Efficient aggregation of encrypted data in wireless sensor networks [C]. Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. San Diego, CA, USA, 2005: 109-117.

[12] Feng T M, Wang C, Zhang W S, Ruan L. Confidentiality protection for distributed sensor data aggregation [C]. Proceedings of the 27th IEEE International Conference on Computer Communications. Phoenix, USA, 2008: 56-60.

[13] Stavros P, Aggelos K, Dimitris P. Secure and efficient in-network processing of exact SUM queries [C]. Proceedings of the 27th International Conference on Data Engineering. Hannover, Germany, 2011: 517-528.

[14] Girao J, Westhoff D, Schneider M. CDA: Concealed data aggregation for reverse multicast traffic in wireless sensor networks [C]. Proceedings of the IEEE International Conference on Communications. Seoul, Korea, 2005: 3044-3049.

[15] Ozdemir S, Yang X. Integrity protecting hierarchical concealed data aggregation for wireless sensor networks [J]. Computer Networks, 2011: 1735-1746.

[16] Groat M M, He W B, Forrest S. KIPDA: *k*-indistinguishable privacy-preserving data aggregation in wireless sensor networks [C]. Proceedings of the 30th IEEE International Conference on Computer Communications. Shanghai, China, 2011: 2024-2032.

[17] Zhang W S, Wang C, Feng T M. GP2S: Generic privacy-preserving solutions for approximate aggregation of sensor data [C]. Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications. Hong Kong, China, 2008: 179-184.

[18] Sheng B, Li Q. Verifiable privacy-preserving range query in two-tiered sensor networks [C]. Proceedings of the 27th IEEE International Conference on Computer Communications. Phoenix, USA, 2008:46-50.

[19] Shi J, Zhang R, Zhang Y C. Secure range queries in tiered sensor networks [C]. Proceedings of the 28th IEEE International Conference on Computer Communications. Rio de Janeiro, Brazil, 2009: 945-953.

[20] Zhang R, Shi J, Zhang Y C. Secure multidimensional range queries in sensor networks [C]. Proceedings of the 10th ACM International symposium on Mobile Ad Hoc Networking and Computing. New Orleans, Louisiana, USA, 2009: 197-206.

[21] Chen F, Liu A X. SafeQ: Secure and efficient query processing in sensor networks [C]. Proceedings of the 29th IEEE International Conference on Computer Communications. San Diego, USA, 2010: 2642-2650.

[22] 范永健, 陈红. 两层传感器网络中可验证隐私保护Top-*k*查询协议 [J]. 计算机学报, 2012,35(3): 423-433.

[23] Nalin S, Yang K, Zhang W S, Qiao D J. ElliPS: A privacy preserving scheme for sensor data storage query [C]. Proceedings of the 28th IEEE International Conference on Computer Communications. Rio de Janeiro, Brazil, 2009: 936-944.

[24] Zhang R, Zhang Y C, Ren K. DP2AC: Distributed privacy-preserving access control in sensor networks [C]. Proceedings of the 28th IEEE International Conference on Computer Communications. Rio de Janeiro, Brazil, 2009: 1251-1259.

[25] He D J, Bu J J , Zhu S C et al. Distributed privacy-preserving access control in a single-owner multi-user sensor networks [C]. Proceedings of the 30th IEEE International Conference on Computer Communications. Shanghai, China, 2011: 331-335.

[26] Hacigumus H, Iyer B R, Li C, Mehrotra S. Executing SQL over encrypted data in the data base service provider model [C]. Proceedings of the ACM SIGMOD International Conference on Management of Data. Madison, Wisconsin, USA, 2002: 216-227.

[27] Hore B, Mehrotra S, Tsudik G. A privacy-preserving index for range queries [C]. Proceedings of the 30th International Conference on Very Large Data Bases. Toronto, Canada, 2004: 720-731.

[28] [Chen](http://www.dblp.org/db/indices/a-tree/c/Chen:Fei.html) F, Liu [A X.](http://www.dblp.org/db/indices/a-tree/l/Liu:Alex_X=.html) . Privacy and integrity-preserving range queries in sensor networks [J]. [IEEE/ACM Transaction on Networks.2012, 20(6):1774-1787.](#ChenL12)

[29] Bozovic V, Socek D, Steinwandt R, et al. Multi-authority attribute-based encryption with honest-but-curious central authority [J]. International Journal of Computer Mathematics. 2012. 89(3): 268-283.

[30] Coman A, Nascimento A M, Sander J. A framework for spatio-temporal query processing over wireless sensor networks [C]. Proceeding of the 1st int workshop on Data management for sensor networks. New York: ACM, 2004:104-110.

[31] Groat M M, Hey W, Forrest S. KIPDA: k-Indistinguishable privacy-preserving data aggregation in wireless sensor networks [C]. Proceeding of 30th IEEE International Conference on Computer Communications. Piscataway, NJ: IEEE, 2011: 2024-2032.