SUPPORT SPATIAL QUERIES WITHOUT CHANGING THE ALGORITHM RUN BY SEMI-TRUSTED THIRD PARTYDARABASE SERVER

M. Radha* & R. Selvakumar**

* PG Scholar, Department of Master of Computer Applications, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu

** Assistant Professor, Department of Master of Computer Applications, Dhanalakshmi Srinivasan Engineering College, Perambalur, Tamilnadu

Abstract:

Location-based services (LBS) require users to continuously report their location to a potentially untrusted server to obtain services based on their location, which can expose them to privacy risks. Unfortunately, existing privacy-preserving techniques for LBS have several limitations, such as requiring a fully-trusted third party, offering limited privacy guarantees and incurring high communication overhead. In this paper, we propose a user-defined privacy grid system called dynamic grid system (DGS); the first holistic system that fulfills four essential requirements for privacy-preserving snapshot and continuous LBS. The system only requires a semi-trusted third party, responsible for carrying out simple matching operations correctly. This semi-trusted third party does not have any information about a user's location. Secure snapshot and continuous location privacy is guaranteed under our defined adversary models. The communication cost for the user does not depend on the user's desired privacy level; it only depends on the number of relevant points of interest in the vicinity of the user. Although we only focus on range and k-nearest-neighbor queries in this work, our system can be easily extended to support other spatial queries without changing the algorithms run by the semi-trusted third party and the database server, provided the required search area of a spatial query can be abstracted into spatial regions. Experimental results show that our DGS is more efficient than the state-of-the-art privacy-preserving technique for continuous LBS.

Introduction:

The increasing availability of location-aware mobile devices has given rise to a flurry of location-based services (LBS). Due to the nature of spatial queries, an LBS needs the user position in order to process her requests. On the other hand, revealing exact user locations to (potentially untrusted) LBS may pinpoint their identities and breach their privacy. To address this issue, spatial anonymity techniques obfuscate user locations, forwarding to the LBS a sufficiently large region instead. Existing methods explicitly target processing in the Euclidean space, and do not apply when proximity to the users is defined according to network distance (e.g., driving time through the roads of a city). The framework model is used for anonymous query processing in networks is proposed. Location obfuscation techniques that (i) provide anonymous LBS access to the users, and (ii) allow efficient query processing at the LBS side is used. The said techniques exploit existing network database infrastructure, requiring no specialized storage schemes or functionalities.

Suitable anonymisation of location data is not a trivial task: under a realistic threat model simply removing explicit identifiers does not anonymise location information. This dissertation describes why this is the case and develops two quantitative security models for anonymising location data: the mix zone model and the variable quality model.

A trusted third-party can use one, or both, models to ensure that all location events given to untrusted applications are suitably anonymised. The mix zone model supports untrusted applications which require accurate location information about users in a set of disjoint physical locations. In contrast, the variable quality model reduces the temporal or spatial accuracy of location information to maintain user anonymity at every location. Both models provide a quantitative measure of the level of anonymity achieved; therefore any given situation can be analysed to determine the amount of information an attacker can gain through analysis of the anonymised data. The suitability of both these models is demonstrated and the level of location privacy available to users of real location-aware applications is measured. Traditionally, computing focused on static home and office scenarios, but mobile and pervasive Computing allow us to move beyond these restricted domains. Weiser noticed these trends over ten years ago and used them to provide some predictions about human-computer interaction in the centu.

Modules Description:

Location-Based Services Awareness:

Traditionally, computers have only had methods of determining and sharing two of the four primitive context types: identity (e.g. authenticating individuals or computers via passwords, digital certificates, etc.) and time (e.g. network time protocol, time, etc.). Location awareness is something which has become important only recently. Knowledge of identity and time have well-defined applications within the traditional (fixed) computing paradigm, whereas location does not. Location information is most useful in combination with mobile and pervasive computing (made possible through the availability of small, portable and cheap hardware), where context information can provide more intuitive human-computer interaction and more aggressive levels of automation.

Query Area:

Location Technologies:

Assessment of accuracy and function is also an essential element of providing a solution to some of the privacy problems presented by both location technologies and their corresponding location-aware applications. In order to locate an entity we must use one or more physical properties of the environment to calculate its position. Many physical properties are amenable to providing location information, but no single technique is suitable for all purposes nor provides all the properties required for every application. Researchers have provided detailed reviews of location technologies in the literature: Hightower provides a survey from the perspective of ubiquitous computing, Azuma analyses the state-of the-art in augmented reality and Welch and Foxlin assess performance of location systems for motion tracking. The Table model outlines the basic accuracy metrics which can be used to measure the performance of a location system. Other important factors include size, weight, installation requirements, robustness to environment (including visual occlusion, heat, sound, magnetic and radio waves), freedom of movement (e.g. wireless operation), power, coverage area and cost. Currently no location system performs well in all cases; application designers must therefore make a decision about what location system best suits their application domain.

Query Request:

A novel query privacy notion in which mobile users are either obligated or willing to reveal their locations, yet they do not want to be identified as the issuer of their location-based queries. Such privacy notion is relaxed from the widely used notion that considers hiding the user location and user query is one process. Several applications can make use of our new notion to enhance the overall quality of location-based services which shows that directly applying existing spatial cloaking techniques to the new query privacy notion would immediately result in two privacy attack models, namely, query sampling and query tracking attacks that can be used by adversaries to infer the actual querying users.

Query Area:

Also identify two main properties, namely, k-sharing region and memorization that if applied to any location cloaking technique, would make it free from the introduced attack models. – We propose a new robust spatial cloaking technique that: (a) distinguishes between location privacy and query privacy, (b) employs the k-sharing region and memorization properties, and (c) supports continuous location-based queries. Furthermore, these techniques require users to trust each other, as they have to reveal their locations to other peers and rely on other peers' locations to blur their locations, another distributed method was proposed that does not require users to trust each other, but it still uses multiple TTPs. Another family of algorithms uses incremental nearest neighbor queries, where a query starts at an "anchor" location which is different from the real location of a user and iteratively retrieves more points of interest until the query is satisfied. While it does not require a trusted third party, the approximate location of a user can still be learned; hence only regional location privacy is achieved. The system provides experimental evidence that the robust spatial cloaking algorithm is scalable in terms of supporting large numbers of users and continuous queries, efficient in terms of supporting various user privacy requirements, provides high-quality services without compromising users' query privacy.

Existing System:

Existing location-based services provide specialized services to their customers based on the knowledge of their exact locations. With untrustworthy servers, location-based services may lead to several privacy threats ranging from worries over employers snooping on their workers' whereabouts to fears of tracking by potential stalkers. While there exist several techniques to preserve location privacy in mobile environments, these techniques are limited as they do not distinguish between location privacy (i.e., a user wants to hide her location) and query privacy (i.e., a user can reveal her location but not her query).This distinction is crucial in many applications where the locations of mobile users are publicly known.

Spatial cloaking techniques have been widely used to preserve user location privacy in LBS. Most of the existing spatial cloaking techniques rely on a fully-trusted third party (TTP), usually termed *location* anonymizer that is required between the user and the service provider. When a user subscribes to LBS, the location anonymizer will blur the user's exact location into a cloaked area such that the cloaked area includes at least k – 1 other users to satisfy k-anonymity. In a system with such *regional location privacy* it is difficult for the user to specify personalized privacy requirements. The feeling based approach alleviates this issue by finding a cloaked area based on the number of its visitors that is at least as popular as the user's specified public region. Although some spatial clocking techniques can be applied to peer-to-peer environments, these techniques still rely on the k-anonymity privacy requirement and can only achieve regional location privacy.

Disadvantages of Existing System:

- ✓ The TTP model has four major drawbacks.
- ✓ It is difficult to find a third party that can be fully trusted.

- ✓ All users need to continuously update their locations with the location anonymizer, even when they are not subscribed to any LBS, so that the location anonymizer has enough information to compute cloaked areas.
- ✓ Because the location anonymizer stores the exact location information of all users, compromising the location anonymizer exposes their locations.
- ✓ k-anonymity typically reveals the approximate location of a user and the location privacy depends on the user distribution.

Reposed System:

In this paper, we go beyond the limitation of existing cloaking algorithms as the proposed new robust spatial cloaking technique for snapshot and continuous location-based queries that clearly distinguishes between location privacy and query privacy. By this distinction, we achieve two main goals: (1) supporting private location-based services to those customers with public locations, and (2) performing spatial cloaking on-demand basis only (i.e., when issuing queries) rather than exhaustively cloaking every single location update. Experimental results show that the robust spatial cloaking algorithm is scalable and efficient while providing anonymity for large numbers of continuous queries without hiding users' locations.

The main idea is to place a semi trusted third party, termed query server (QS), between the user and the service provider (SP). QS Only needs to be semi-trusted because it will not collect/store or even have access to any user location information. Semi-trusted in this context means that while QS will try to determine the location of a user, it still correctly carries out the simple matching operations required in the protocol, i.e., it does not modify or drop messages or create new messages. An untrusted QS would arbitrarily modify and drop messages as well as inject fake messages, which is why our system depends on a semi-trusted QS.

The main idea is to place a semi trusted third party, termed *query server* (QS), between the user and the service provider (SP). QS only needs to be semi-trusted because it will not collect/store or even have access to any user location information. *Semi-trusted* in this context means that while QS will try to determine the location of a user, it still correctly carries out the simple matching operations required in the protocol, i.e., it does not modify or drop messages or create new messages. An untrusted QS would arbitrarily modify and drop messages as well as inject fake messages, which is why our system depends on a semi-trusted QS.

Advantages of Proposed System:

For each selected POI, SP encrypts its information, using the dynamic grid structure specified by the user to find a grid cell covering the POI, and encrypts the cell identity to produce the encrypted identifier for that POI. The encrypted POIs with their corresponding encrypted identifiers are returned to QS. QS stores the set of encrypted POIs and only returns to the user a subset of encrypted POIs whose corresponding identifiers match any one of the encrypted identifiers initially sent by the user. After the user receives the encrypted POIs, she decrypts them to get their exact locations and computes a query answer.

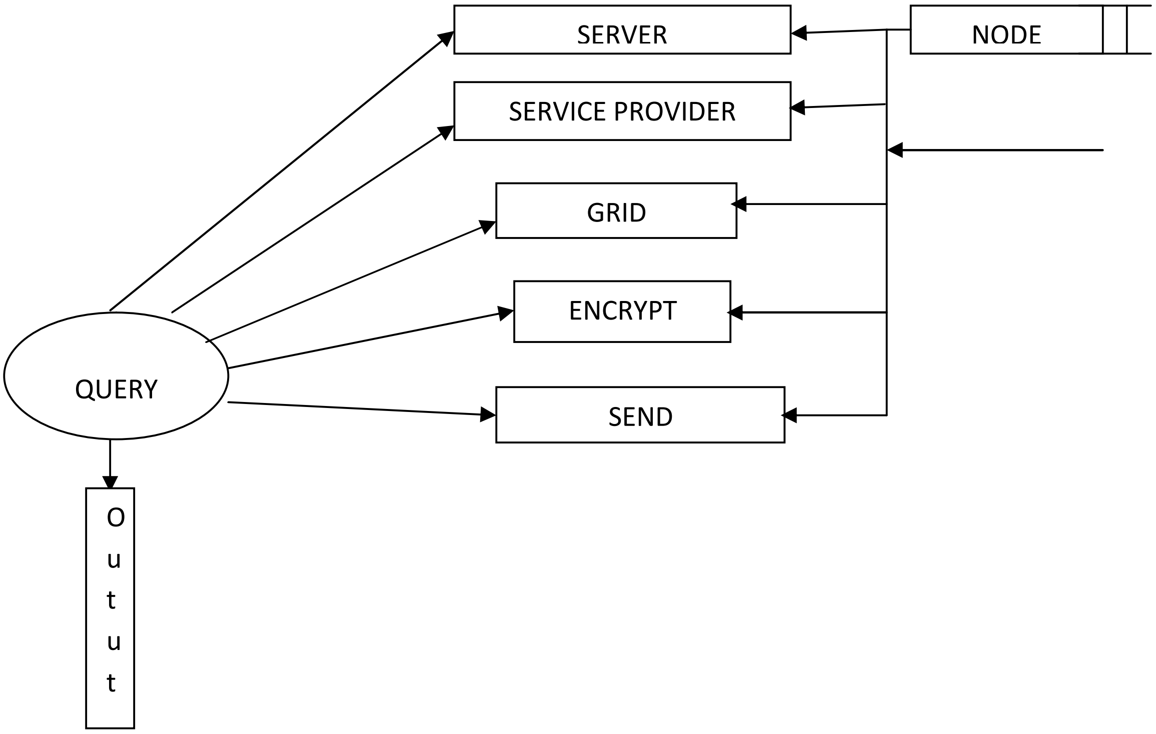
System Implementation:

Implementation is the process of converting a new or revised system design into an operational one. The implementation is the final and important phase. It involves the training, system testing and successfully running of developed proposed system. The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of developed system using various kinds of data.

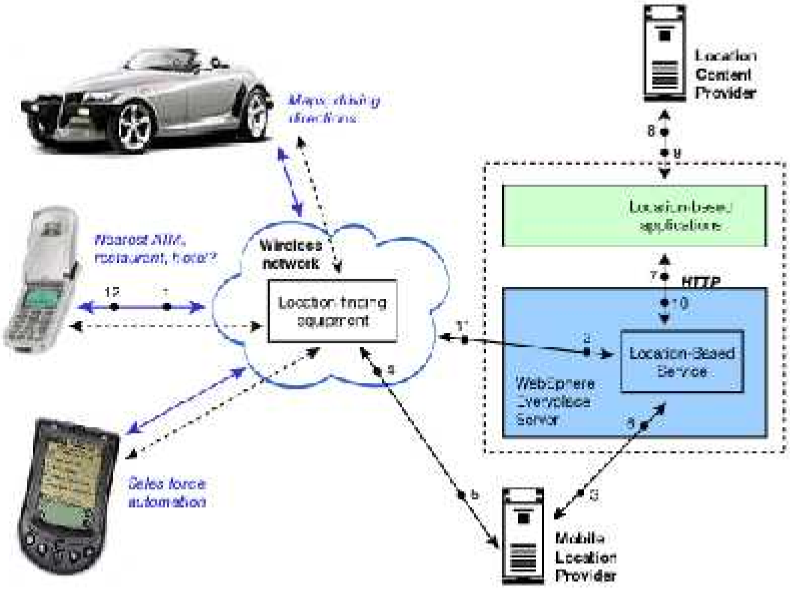
An elaborate testing of data is prepared and the system is tested using that test data. The corrections are also noted for future use. The users are trained to operate the developed system. Both the hardware and software securities are made to run the developed system successfully in future.

Implementation is the process of converting a new or revised system design in to an operational one. Education of user should really have taken place much earlier in the project when they were being involved in the investigation and design work. Training has to be given to the user regarding the new system. Once the user has been trained, the system can be tested hardware and software securities are to run the developed system successfully in the future.

Data Flow Diagrams:



System Architecture:


Location Content Provider
8
9
Maps, driving directions
Nearest ATM, restaurant, hotel?
Wireless network
Location tracing equipment
12
1
11
2
4
Location-based applications
7
HTTP
10
Location-Based Service
WebSphere Everyplace Server
6
5
3
Sales force automation
Mobile Location Provider

Requirements Gathering:

The first phase of software project is to gather requirements. Gathering software requirements begins as a creative brainstorming process in which the goal is to develop an idea for a new product that no other software vendor has thought. New software product ideas normally materialize as a result of analyzing market data and interviewing customers about their product needs.

The main function of the requirements gathering phase is to take an abstract idea that fills a particular need or that solves a particular problem and create a real world project with a particular set of objectives, a budget, a timeline and a team.

- ✓ Some of the highlights of the requirement gathering phase include
- ✓ Collecting project ideas
- ✓ Gathering customer requirements and proposed solutions
- ✓ Justifying the project
- ✓ Submitting the request for proposal
- ✓ Getting the team in place
- ✓ Preparing the requirements document

Collecting Project Ideas:

Coming up with project ideas can prove expansion exercise.

Gathering Customer Requirements and Proposed Solutions:

Focus on solving the customer’s particular problems and fulfilling the customer’s needs.

Justifying the Project:

An important part of the requirements gathering phase is to decide whether a particular project is more or less likely to make the company successful.

Submitting the Request for Proposal:

The request for proposal is an important document that senior management uses to decide which projects are likely to show a profit and therefore are worthy of allocating a budget for further testing and study.

Getting the Team in Place:

One of the requirements of every project is a team to work on the project. The team includes the project sponsor, a project manager, analysts, developers, database administrators, technical writers, quality assurance testers, trainers, release managers and possibly others.

Preparing the Requirements Document:

The Requirements Document (RD) becomes main input to analysis phase. It includes a summary of the requirements to solve the objective of the project, the major features of the product, a documentation plan, a support plan, and licensing issues.

System Maintenance:

Enabling Privacy for the Paranoids:

P3P is a set of standards that allow corporations to declare their privacy policies. Hippocratic Databases have been proposed to implement such policies within a corporation's data store. From an end-user individual's point of view, both of these rest on an uncomfortable philosophy of trusting corporations to protect his/her privacy. Recent history chronicles several episodes when such trust has been willingly or accidentally violated by corporations facing bankruptcy courts, civil subpoenas or lucrative mergers.

We contend that data management solutions for information privacy must restore controls in the individual's hands. We suggest that enabling such control will require a radical re-think on modeling, release/acquisition, and management of personal data.

Location Privacy in Ubiquitous Computing:

The field of ubiquitous computing envisages an era when the average consumer owns hundreds or thousands of mobile and embedded computing devices. These devices will perform actions based on the context of their users, and therefore ubiquitous systems will gather, collate and distribute much more personal information about individuals than computers do today.

Much of this personal information will be considered private, and therefore mechanisms which allow users to control the dissemination of these data are vital. Location information is a particularly useful form of context in ubiquitous computing, yet its unconditional distribution can be very invasive.

Benchmarking spatiotemporal database systems requires the definition of suitable datasets simulating the typical behavior of moving objects. Previous approaches for generating spatiotemporal data do not consider that moving objects often follow a given network. Therefore, benchmarks require datasets consisting of such "network-based" moving objects. In this paper, the most important properties of network-based moving objects are presented and discussed. Essential aspects are the maximum speed and the maximum capacity of connections, the influence of other moving objects on the speed and the route of an object, the adequate determination of the start and destination of an object, the influence of external events, and time-scheduled traffic.

These characteristics are the basis for the specification and development of a new generator for spatiotemporal data. This generator combines real data (the network) with user-defined properties of the resulting dataset. A framework is proposed where the user can control the behavior of the generator by re-defining the functionality of selected object classes. An experimental performance investigation demonstrates that the chosen approach is suitable for generating large data sets.

This paper tackles a major privacy threat in current location-based services where users have to report their exact locations to the database server in order to obtain their desired services. For example, a mobile user asking about her nearest restaurant has to report her exact location. With untrusted service providers, reporting private location information may lead to several privacy threats. In this paper, we present a peer-to-peer (P2P) spatial cloaking algorithm in which mobile and stationary users can entertain location-based services without revealing their exact location information.

The main idea is that before requesting any location-based service, the mobile user will form a group from her peers via single-hop communication and/or multi-hop routing. Then, the spatial cloaked area is computed as the region that covers the entire group of peers. Two modes of operations are supported within the proposed P2P spatial cloaking algorithm, namely, the on-demand mode and the proactive mode. Experimental results show that the P2P spatial cloaking algorithm operated in the on-demand mode has lower communication cost and better quality of services than the proactive mode, but the on-demand incurs longer response time.

Obfuscation concerns the practice of deliberately degrading the quality of information in some way, so as to protect the privacy of the individual to whom that information refers. In this paper, we argue that obfuscation is an important technique for protecting an individual's location privacy within a pervasive computing environment. The paper sets out a formal framework within which obfuscated location-based services are defined.

This framework provides a computationally efficient mechanism for balancing an individual's need for high-quality information services against that individual's need for location privacy. Negotiation is used to ensure that a location-based service provider receives only the information it needs to know in order to provide a service of satisfactory quality. The results of this work have implications for numerous applications of mobile and location-aware systems, as they provide a new theoretical foundation for addressing the privacy concerns that are acknowledged to be retarding the widespread acceptance and use of location-based services.

Requirements Gathering:

The first phase of software project is to gather requirements. Gathering software requirements begins as a creative brainstorming process in which the goal is to develop an idea for a new product that no other software vendor has thought. New software product ideas normally materialize as a result of analyzing market data and interviewing customers about their product needs.

The main function of the requirements gathering phase is to take an abstract idea that fills a particular need or that solves a particular problem and create a real world project with a particular set of objectives, a budget, a timeline and a team.

- ✓ Some of the highlights of the requirement gathering phase include
- ✓ Collecting project ideas
- ✓ Gathering customer requirements and proposed solutions
- ✓ Justifying the project
- ✓ Submitting the request for proposal
- ✓ Getting the team in place
- ✓ Preparing the requirements document

Collecting Project Ideas:

Coming up with project ideas can prove expansion exercise.

Gathering Customer Requirements and Proposed Solutions:

Focus on solving the customer's particular problems and fulfilling the customer's needs.

Justifying the Project:

An important part of the requirements gathering phase is to decide whether a particular project is more or less likely to make the company successful.

Submitting the Request for Proposal:

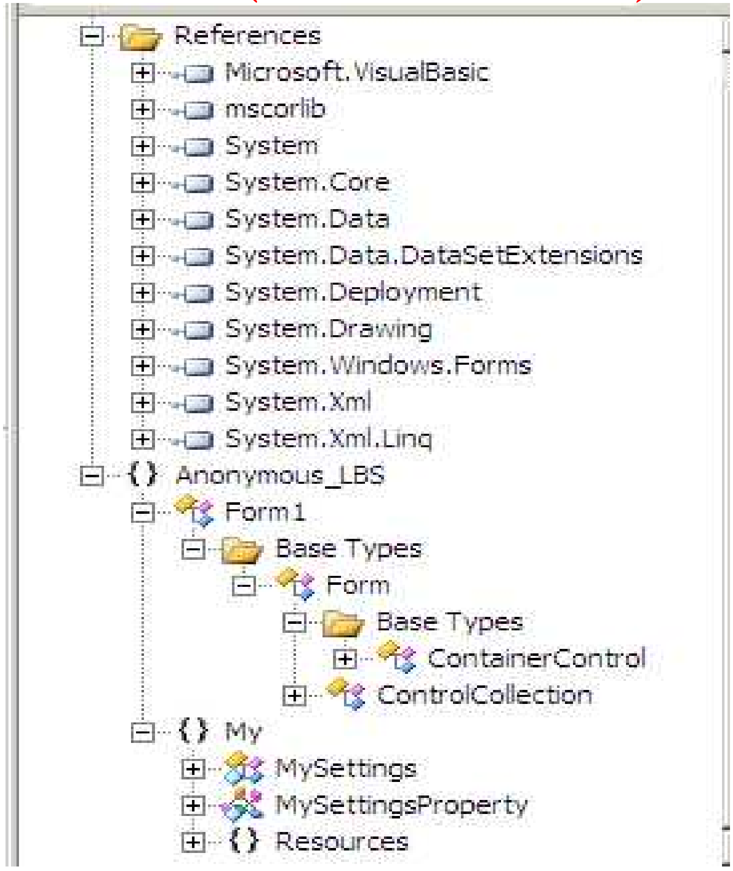
The request for proposal is an important document that senior management uses to decide which projects are likely to show a profit and therefore are worthy of allocating a budget for further testing and study.

Getting the Team in Place:

One of the requirements of every project is a team to work on the project. The team includes the project sponsor, a project manager, analysts, developers, database administrators, technical writers, quality assurance testers, trainers, release managers and possibly others.

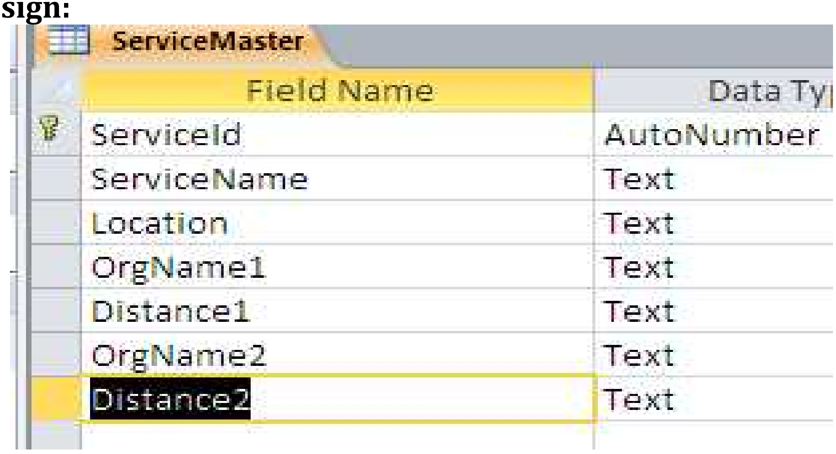
Preparing the Requirements Document:

The Requirements Document (RD) becomes main input to analysis phase. It includes a summary of the requirements to solve the objective of the project, the major features of the product, a documentation plan, a support plan, and licensing issues.



```
References
    Microsoft.VisualBasic
    mscorlib
    System
    System.Core
    System.Data
    System.Data.DataSetExtensions
    System.Deployment
    System.Drawing
    System.Windows.Forms
    System.Xml
    System.Xml.Linq
{} Anonymous_LBS
    Form1
        Base Types
            Form
                Base Types
                    ContainerControl
                ControlCollection
    {} My
        MySettings
        MySettingsProperty
        {} Resources
```

Database Design:



ServiceMaster

Field Name	Data Ty
ServiceId	AutoNumber
ServiceName	Text
Location	Text
OrgName1	Text
Distance1	Text
OrgName2	Text
Distance2	Text

Conclusion:

DTN technologies are becoming successful solutions in military applications that allow wireless devices to communicate with each other and access the confidential information reliably by exploiting external storage nodes. DES is a scalable cryptographic solution to the access control and secure data retrieval issues. In this paper, we proposed an efficient and secure data retrieval method using DES for decentralized DTNs where multiple key authorities manage their attributes independently. The inherent key escrow problem is resolved such that the confidentiality of the stored data is guaranteed even under the hostile environment where key authorities might be compromised or not fully trusted. In addition, the fine-grained key revocation can be done for each attribute group. We demonstrate how to apply the proposed mechanism to securely and efficiently manage the confidential data distributed in the disruption-tolerant military network.

References:

1. J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “MaxProp: Routing for vehicle-based disruption-tolerant networking,” in Proc. IEEE Conf. Comput. Commun., Barcelona, Spain, Apr. 2006, pp. 1–11.
2. P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, “Pocket switched networks and the consequences of human mobility in conference environments,” in Proc. ACM SIGCOMM Workshop Delay Tolerant Netw., Philadelphia, PA, USA, Aug. 2005,pp. 244–251.
3. D. Zhao, H. Ma, S. Tang, et al., “COUPON: A cooperative framework for building sensing maps in mobile opportunistic networks,” to appear in IEEE Trans. Parallel Distrib. Syst., Feb.2014.
4. A. Seth, D. Kroeker, M. Zaharia, S. Guo, and S. Keshav, “Low-cost communication for rural internet kiosks using mechanical backhaul,” in Proc. 12th Annu. ACM Int. Conf. Mobile Comput. Netw., Los Angeles, CA, USA, Sep. 2006, pp. 334–345.
5. M. Grossglauser and D. N. C. Tse, “Mobility increases the capacity of ad hoc wireless networks,” IEEE/ACM Trans. Netw., vol. 10, no. 4, pp. 477–486, Aug. 2002.
6. P. Gupta and P. R. Kumar, “The capacity of wireless networks,” IEEE Trans. Inf. Theory, vol. 46, no. 2, pp. 388–404, Mar. 2000.
7. P. Li, Y. Fang, J. Li, and X. Huang, “Smooth trade-offs between throughput and delay in mobile ad hoc networks,” IEEE Trans. Mobile Comput., vol. 11, no. 3, pp. 427–438, Mar. 2012.
8. D. Ciullo, V. Martina, M. Garetto, and E. Leonardi, “Impact of correlated mobility on delay-throughput performance in mobile ad hoc networks,” IEEE/ACM Trans. Netw., vol. 19, no. 6, pp. 1745–1758, Dec. 2011.