# Design and Development of an Effective Mechanism for Secure Management of Confidential Data in Disruption Tolerant Military Network

**M. Swarna Latha[1]; A. Shobha Rani[2]& Prof.Dr.G.Manoj Someswar[3]**

[1]M.Tech.(CSE) from Narasimha Reddy Engineering College, Affiliated to JNTUH, Hyderabad, Telangana, India

[2]M.Tech. (CSE), Assistant Professor, Department of CSE, Narasimha Reddy Engineering College, Affiliated to JNTUH, Hyderabad, Telangana, India

[3]B.Tech., M.S.(USA), M.C.A., Ph.D., Principal & Professor, Department Of CSE, Anwar-ul-uloom College of Engineering & Technology, Affiliated to JNTUH, Vikarabad, Telangana, India

**ABSTRACT:**

*Mobile nodes in military environments such as a battlefield or a hostile region are likely to suffer from intermittent network connectivity and frequent partitions. Disruption-tolerant network (DTN) technologies are becoming successful solutions that allow wireless devices carried by soldiers to communicate with each other and access the confidential information or command reliably by exploiting external storage nodes. Some of the most challenging issues in this scenario are the enforcement of authorization policies and the policies update for secure data retrieval. Ciphertext-policy attribute-based encryption (CP-ABE) is a promising cryptographic solution to the access control issues. However, the problem of applying CP-ABE in decentralized DTNs introduces several security and privacy challenges with regard to the attribute revocation, key escrow, and coordination of attributes issued from different authorities. In this paper, we propose a secure data retrieval scheme using CP-ABE for decentralized DTNs where multiple key authorities manage their attributes independently. We demonstrate how to apply the proposed mechanism to securely and efficiently manage the confidential data distributed in the disruption-tolerant military network.*

**KEYWORDS:** Node-Density Based Adaptive Routing (NDBAR); Ciphertext Policy Attributed-Based Encryption (CP-ABE); L-hop Neighborhood Spraying (LNS); Highest Encounter First Routing (HEFR); Attribute-Based Encryption (ABE*)*

## INTRODUCTION

Networking is the word basically relating to computers and their connectivity. It is very often used in the world of computers and their use in different connections. The term networking implies the link between two or more computers and their devices, with the vital purpose of sharing the data stored in the computers, with each other. The networks between the computing devices are very common these days due to the launch of various hardware and computer software which aid in making the activity much more convenient to build and use.

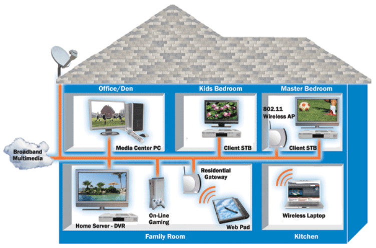

**Figure 1: Structure of Networking between the different computers**

**General Network Techniques** - When computers communicate on a network, they send out data packets without knowing if anyone is listening. Computers in a network all have a connection to the network and that is called to be connected to a network bus. What one computer sends out will reach all the other computers on the local network.

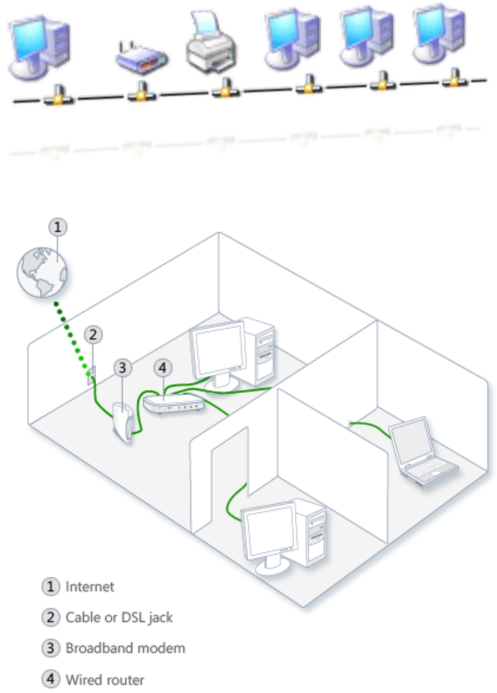

**Figure 2 : Above diagrams show the clear idea about the networking functions**

For the different computers to be able to distinguish between each other, every computer has a unique ID called MAC-address (Media Access Control Address). This address is not only unique on your network but unique for all devices that can be hooked up to a network. The MAC-address is tied to the hardware and has nothing to do with IP-addresses. Since all computers on the network receives everything that is sent out from all other computers the MAC-addresses is primarily used by the computers to filter out incoming network traffic that is addressed to the individual computer.

When a computer communicates with another computer on the network, it sends out both the other computers MAC-address and the MAC-address of its own. In that way the receiving computer will not only recognize that this packet is for me but also, who sent this data packet so a return response can be sent to the sender.

**On an Ethernet network** as described here, all computers hear all network traffic since they are connected to the same bus. This network structure is called multi-drop.

One problem with this network structure is that when you have, let say ten (10) computers on a network and they communicate frequently and due to that they sends out there data packets randomly, collisions occur when two or more computers sends data at the same time. When that happens data gets corrupted and has to be resent. On a network that is heavy loaded even the resent packets collide with other packets and have to be resent again. In reality this soon becomes a bandwidth problem. If several computers communicate with each other at high speed they may not be able to utilize more than 25% of the total network bandwidth since the rest of the bandwidth is used for resending previously corrupted packets. The way to minimize this problem is to use network switches.

**Characteristics of Networking:**

The following characteristics should be considered in network design and ongoing maintenance:

1) **Availability**is typically measured in a percentage based on the number of minutes that exist in a year. Therefore, uptime would be the number of minutes the network is available divided by the number of minutes in a year.

2) **Cost** includes the cost of the network components, their installation, and their ongoing maintenance.

3) **Reliability** defines the reliability of the network components and the connectivity between them. Mean time between failures (MTBF) is commonly used to measure reliability.

4) **Security**includes the protection of the network components and the data they contain and/or the data transmitted between them.

5) **Speed** includes how fast data is transmitted between network end points (the data rate).

6) **Scalability** defines how well the network can adapt to new growth, including new users, applications, and network components.

7) **Topology** describes the physical cabling layout and the logical way data moves between components.

**Types of Networks:**

Organizations of different structures, sizes, and budgets need different types of networks. Networks can be divided into one of two categories:

- peer-to-peer
- server-based networks

1. **Peer-to-Peer Network:**

A peer-to-peer network has no dedicated servers; instead, a number of workstations are connected together for the purpose of sharing information or devices. Peer-to-peer networks are designed to satisfy the networking needs of home networks or of small companies that do not want to spend a lot of money on a dedicated server but still want to have the capability to share information or devices like in school, college, cyber cafe

2. **Server-Based Networks:**

In server-based network data files that will be used by all of the users are stored on the one server. With a server-based network, the network server stores a list of users who may use network resources and usually holds the resources as well.
This will help by giving you a central point to set up permissions on the data files, and it will give you a central point from which to back up all of the data in case data loss should occur.

**Network Communications:**

- Computer networks use signals to transmit data, and protocols are the languages computers use to communicate.
- Protocols provide a variety of communications services to the computers on the network.
- Local area networks connect computers using a shared, half-duplex, baseband medium, and wide area networks link distant networks.
- Enterprise networks often consist of clients and servers on horizontal segments connected by a common backbone, while peer-to-peer networks consist of a small number of computers on a single LAN.

**Advantages of Networking:**

**1. Easy Communication:**

It is very easy to communicate through a network. People can communicate efficiently using a network with a group of people. They can enjoy the benefit of emails, instant messaging, telephony, video conferencing, chat rooms, etc.

**2. Ability to Share Files, Data and Information:**

This is one of the major advantages of networking computers. People can find and share information and data because of networking. This is beneficial for large organizations to maintain their data in an organized manner and facilitate access for desired people.

**3. Sharing Hardware:**

Another important advantage of networking is the ability to share hardware. For an example, a printer can be shared among the users in a network so that there's no need to have individual printers for each and every computer in the company. This will significantly reduce the cost of purchasing hardware.

**4. Sharing Software:**

Users can share software within the network easily. Networkable versions of software are available at considerable savings compared to individually licensed version of the same software. Therefore large companies can reduce the cost of buying software by networking their computers.

**5. Security:**

Sensitive files and programs on a network can be password protected. Then those files can only be accessed by the authorized users. This is another important advantage of networking when there are concerns about security issues. Also each and every user has their own set of privileges to prevent those accessing restricted files and programs.

**6. Speed:**

Sharing and transferring files within networks is very rapid, depending on the type of network. This will save time while maintaining the integrity of files.

## LITERATURE SURVEY

Disruption-tolerant networks (DTNs) attempt to route network messages via intermittently connected nodes. Routing in such environments is difficult because peers have little information about the state of the partitioned network and transfer opportunities between peers are of limited duration. In this paper, we propose MaxProp, a protocol for effective routing of DTN messages. MaxProp is based on prioritizing both the schedule of packets transmitted to other peers and the schedule of packets to be dropped. These priorities are based on the path likelihoods to peers according to historical data and also on several complementary mechanisms, including acknowledgments, a head-start for new packets, and lists of previous intermediaries. Our evaluations show that MaxProp performs better than protocols that have access to an oracle that knows the schedule of meetings between peers. Our evaluations are based on 60 days of traces from a real DTN network we have deployed on 30 buses. Our network, called UMassDieselNet, serves a large geographic area between five colleges. We also evaluate MaxProp on simulated topologies and show it performs well in a wide variety of DTN environments.

Traditional ad hoc routing protocols do not work in intermittently connected networks since end-to-end paths may not exist in such networks. Hence, routing mechanisms that can withstand disruptions need to he designed. A store-and-forward approach has been proposed for disruption tolerant networks. Recently, several approaches have been proposed for unicast routing in disruption-prone networks e.g. the 2-hop relay approach, delivery probability based routing, and message ferrying. In our earlier paper, we have evaluated a combined multihop and message ferrying approach in disruption tolerant networks. In that paper, we assume that a special node is designated to be a message ferry. A more flexible approach is to let regular nodes volunteer to be message ferries when network dynamics mandate the presence of such ferries to ensure communications. Thus, in this paper, we design a node-density based adaptive routing (NDBAR) scheme that allows regular nodes to volunteer to be message ferries when there are very few nodes around them to ensure the feasibility of continued communications. Our simulation results indicate that our NDBAR scheme can achieve the highest delivery ratio in very sparse networks that are prone to frequent disruptions

Message ferrying is a networking paradigm where a special node, called a message ferry, facilitates the connectivity in a mobile ad hoc network where the nodes are sparsely deployed. One of the key challenges under this paradigm is the design of ferry routes to achieve certain properties of end-to-end connectivity, such as, delay and message loss among the nodes in the ad hoc network. This is a difficult problem when the nodes in the network move arbitrarily. As we cannot be certain of the location of the nodes, we cannot design a route where the ferry can contact the nodes with certainty. Due to this difficulty, prior work has either considered ferry route design for ad hoc networks where the nodes are stationary, or where the nodes and the ferry move pro-actively in order to meet at certain locations. Such systems either require long-range radio or disrupt nodes' mobility patterns which can be dictated by non-communication tasks. We present a message ferry route design algorithm that we call the Optimized Way-points, or OPWP, that generates a ferry route which assures good performance without requiring any online collaboration between the nodes and the ferry. The OPWP ferry route comprises a set of way-points and waiting times at these way-points, that are chosen carefully based on the node mobility model. Each time that the ferry traverses this route, it contacts each mobile node with a certain minimum probability. The node-ferry contact probability in turn determines the frequency of node-ferry contacts and the properties of end-to-end delay. We show that OPWP consistently outperforms other naive ferry routing approaches.

Mobile Nodes in some challenging network scenarios suffer from intermittent connectivity and frequent partitions e.g. battlefield and disaster recovery scenarios. Disruption Tolerant Network (DTN) technologies are designed to enable nodes in such environments to communicate with one another. Several application scenarios require a security design that provides fine grain access control to contents stored in storage nodes within a DTN or to contents of the messages routed through the network. In this paper, we propose an access control scheme which is based on the Ciphertext Policy Attributed-Based Encryption (CP-ABE) approach. Our scheme provides a flexible fine-grained access control such that the encrypted contents can only be accessed by authorized users. Two unique features our scheme provide are: (i) the incorporation of dynamic attributes whose value may change over time, and (ii) the revocation feature. We also provide some performance results from our implementation.

Mobile nodes in some challenging network scenarios suffer from intermittent connectivity and frequent partitions e.g. battlefield and disaster recovery scenarios. Disruption Tolerant Network (DTN) technologies are designed to enable nodes in such environments to communicate with one another. Several DTN routing schemes have been proposed. However, not much work has been done on providing information access in such challenging network scenarios. Existing client/server paradigm for information access will not be feasible in such scenarios since end-to-end path does not exist. Thus, in this paper, we explore how a content-based information retrieval system can be designed for DTNs. There are three important design issues, namely (a) how should data be replicated and stored at multiple nodes, (b) how should a query be disseminated in sparsely connected networks, (c) how should a query response be routed back to the querying node. We first describe two data caching schemes: (a) K-copy random caching, (b) K-copy intelligent caching. Then, we describe an L-hop Neighborhood Spraying (LNS) scheme for query dissemination. For message routing, we either use Prophet routing scheme or Highest Encounter First Routing (HEFR) scheme. We conduct extensive simulation studies to evaluate different combinations of these algorithms. Our results reveal that the scheme that performs the best is the one that uses the K-copy intelligent caching combined with the LNS query dissemination and HEFR scheme.

## SYSTEM STUDY

## FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

## ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

## TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

## SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

## SYSTEM DESIGN

## SYSTEM ARCHITECTURE:

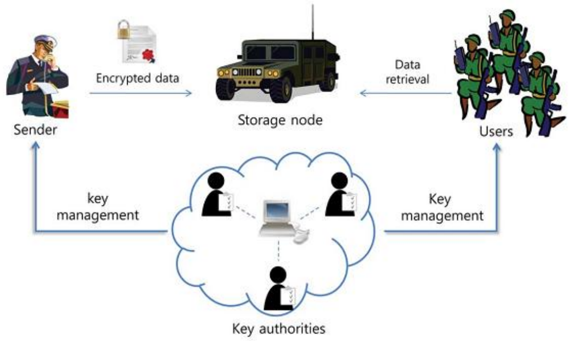

**Figure 3 : System Architecture**

## DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

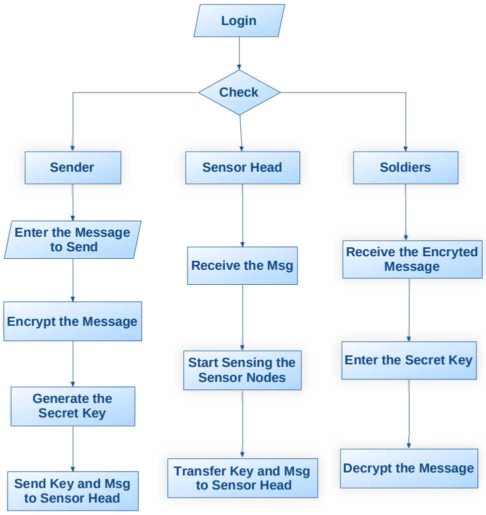
Login
Check
Sender
Sensor Head
Soldiers
Enter the Message to Send
Receive the Msg
Receive the Encryted Message
Encrypt the Message
Start Sensing the Sensor Nodes
Enter the Secret Key
Generate the Secret Key
Send Key and Msg to Sensor Head
Transfer Key and Msg to Sensor Head
Decrypt the Message

**Figure 4 : Data Flow Diagram**

## UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

## GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.

5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

## USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

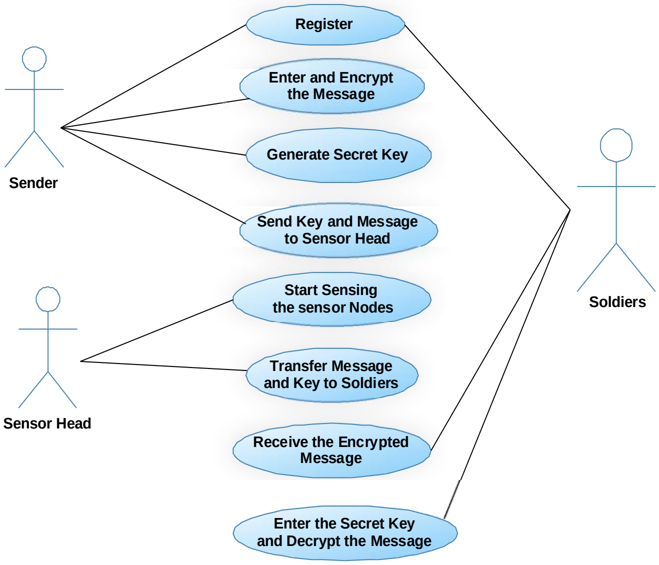
Register
Enter and Encrypt the Message
Generate Secret Key
Send Key and Message to Sensor Head
Start Sensing the sensor Nodes
Transfer Message and Key to Soldiers
Receive the Encrypted Message
Enter the Secret Key and Decrypt the Message
Sender
Sensor Head
Soldiers

**Figure 5 : Use Case Diagram**

## CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

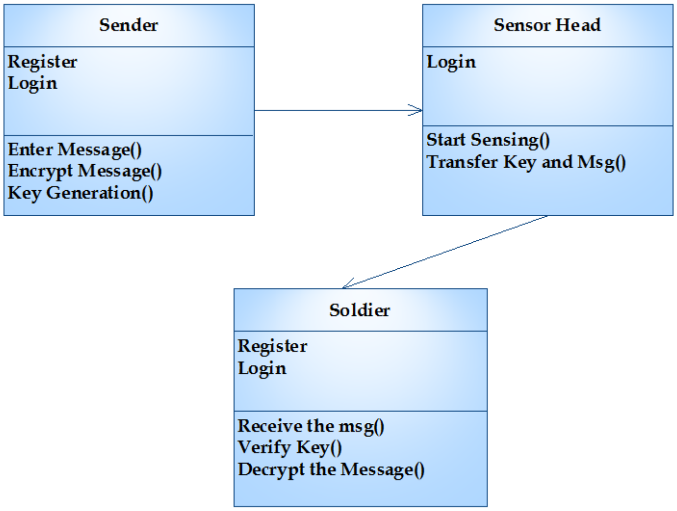

**Figure 6 : Class Diagram**

## SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

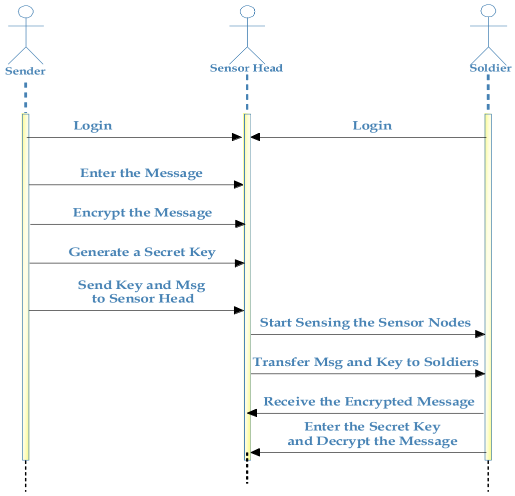
Sender
Sensor Head
Soldier
Login
Login
Enter the Message
Encrypt the Message
Generate a Secret Key
Send Key and Msg to Sensor Head
Start Sensing the Sensor Nodes
Transfer Msg and Key to Soldiers
Receive the Encrypted Message
Enter the Secret Key and Decrypt the Message

**Figure 7: Sequence Diagram**

## ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

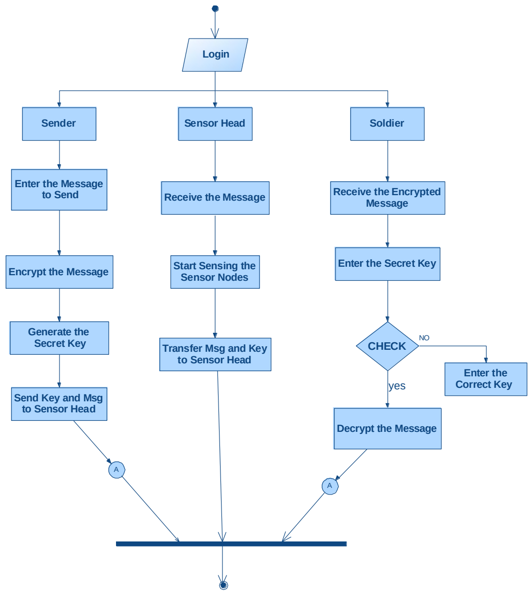

**Figure 8: Activity Digram**

## INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

## OBJECTIVES

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2.It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3.When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

## OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2.Select methods for presenting information.

3.Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- ❖ Convey information about past activities, current status or projections of the
- ❖ Future.
- ❖ Signal important events, opportunities, problems, or warnings.
- ❖ Trigger an action.
- ❖ Confirm an action.

## SYSTEM ANALYSIS

### EXISTING SYSTEM:

The concept of attribute-based encryption (ABE) is a promising approach that fulfills the requirements for secure data retrieval in DTNs. ABE features a mechanism that enables an access control over encrypted data using access policies and ascribed attributes among private keys and ciphertexts. Especially, ciphertext-policy ABE (CP-ABE) provides a scalable way of encrypting data such that the encryptor defines the attribute set that the decryptor needs to possess in order to decrypt the ciphertext. Thus, different users are allowed to decrypt different pieces of data per the security policy.

### DISADVANTAGES OF EXISTING SYSTEM:

- ♣ The problem of applying the ABE to DTNs introduces several security and privacy challenges. Since some users may change their associated attributes at some point (for example, moving their region), or some private keys might be compromised, key revocation (or update) for each attribute is necessary in order to make systems secure.
- ♣ However, this issue is even more difficult, especially in ABE systems, since each attribute is conceivably shared by multiple users (henceforth, we refer to such a collection of users as an attribute group)
- ♣ Another challenge is the key escrow problem. In CP-ABE, the key authority generates private keys of users by applying the authority's master secret keys to users' associated set of attributes.
- ♣ The last challenge is the coordination of attributes issued from different authorities. When multiple authorities manage and issue attributes keys to users independently with their own master secrets, it is very hard to define fine-grained access policies over attributes issued from different authorities.

### PROPOSED SYSTEM:

In this paper, we propose an attribute-based secure data retrieval scheme using CP-ABE for decentralized DTNs. The proposed scheme features the following achievements. First, immediate attribute revocation enhances backward/forward secrecy of confidential data by reducing the windows of vulnerability. Second, encryptors can define a fine-grained access policy using any monotone access structure under attributes issued from any chosen set of authorities. Third, the key escrow problem is resolved by an escrow-free key issuing protocol that exploits the characteristic of the decentralized DTN architecture. The key issuing protocol generates and issues user secret keys by performing a secure two-party computation (2PC) protocol among the key authorities with their own master secrets. The 2PC protocol deters the key authorities from obtaining any master secret information of each other such that none of them could generate the whole set of user keys alone. Thus, users are not required to fully trust the authorities in order to protect their data to be shared. The data confidentiality and privacy can be cryptographically enforced against any curious key authorities or data storage nodes in the proposed scheme.

## ADVANTAGES OF PROPOSED SYSTEM:

- **Data confidentiality:** Unauthorized users who do not have enough credentials satisfying the access policy should be deterred from accessing the plain data in the storage node. In addition, unauthorized access from the storage node or key authorities should be also prevented.
- **Collusion-resistance:** If multiple users collude, they may be able to decrypt a ciphertext by combining their attributes even if each of the users cannot decrypt the ciphertext alone.
- **Backward and forward Secrecy:** In the context of ABE, backward secrecy means that any user who comes to hold an attribute (that satisfies the access policy) should be prevented from accessing the plaintext of the previous data exchanged before he holds the attribute. On the other hand, forward secrecy means that any user who drops an attribute should be prevented from accessing the plaintext of the subsequent data exchanged after he drops the attribute, unless the other valid attributes that he is holding satisfy the access policy.

## SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

## TYPES OF TESTS

### Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

### Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

### Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

### System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

### White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

### Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

### Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

### Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

### Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

### Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

### Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

### Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## IMPLEMENTATION

### MODULES:

1. Key Authorities
2. Storage Nodes
3. Sender
4. User

### MODULES DESCRIPTION:

### Key Authorities:

They are key generation centers that generate public/secret parameters for CP-ABE. The key authorities consist of a central authority and multiple local authorities. We assume that there are secure and reliable communication channels between a central authority and each local authority during the initial key setup and generation phase. Each local authority manages different attributes and issues corresponding attribute keys to users. They grant differential access rights to individual users based on the users' attributes. The key authorities are assumed to be honest-but-curious. That is, they will honestly execute the assigned tasks in the system; however they would like to learn information of encrypted contents as much as possible.

### Storage node:

This is an entity that stores data from senders and provide corresponding access to users. It may be mobile or static. Similar to the previous schemes, we also assume the storage node to be semi-trusted that is honest-but-curious.

### Sender:

This is an entity who owns confidential messages or data (e.g., a commander) and wishes to store them into the external data storage node for ease of sharing or for reliable delivery to users in the extreme networking environments. A sender is responsible for defining (attribute based) access policy and enforcing it on its own data by encrypting the data under the policy before storing it to the storage node.

### User:

This is a mobile node who wants to access the data stored at the storage node (e.g., a soldier). If a user possesses a set of attributes satisfying the access policy of the encrypted data defined by the sender, and is not revoked in any of the attributes, then he will be able to decrypt the ciphertext and obtain the data.

## RESULTS &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. CP-ABE 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 CP-ABE 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 networks," in *Proc. IEEE INFOCOM*, 2006, pp. 1–11.

[2] M. Chuah and P. Yang, "Node density-based adaptive routing scheme for disruption tolerant networks," in *Proc. IEEE MILCOM*, 2006, pp. 1–6.

[3] M. M. B. Tariq, M. Ammar, and E. Zequra, "Mesage ferry route design for sparse ad hoc networks with mobile nodes," in *Proc. ACM MobiHoc*, 2006, pp. 37–48.

[4] S. Roy andM. Chuah, "Secure data retrieval based on ciphertext policy attribute-based encryption (CP-ABE) system for the DTNs," Lehigh CSE Tech. Rep., 2009.

[5] M. Chuah and P. Yang, "Performance evaluation of content-based information retrieval schemes for DTNs," in *Proc. IEEE MILCOM*, 2007, pp. 1–7.

[6] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, "Plutus: Scalable secure file sharing on untrusted storage," in *Proc. Conf. File Storage Technol.*, 2003, pp. 29–42.

[7] L. Ibraimi, M. Petkovic, S. Nikova, P. Hartel, and W. Jonker, "Mediated ciphertext-policy attribute-based encryption and its application," in *Proc. WISA*, 2009, LNCS 5932, pp. 309–323.

[8] N. Chen, M. Gerla, D. Huang, and X. Hong, "Secure, selective group broadcast in vehicular networks using dynamic attribute based encryption," in *Proc. Ad Hoc Netw. Workshop*, 2010, pp. 1–8.

[9] D. Huang and M. Verma, "ASPE: Attribute-based secure policy enforcement in vehicular ad hoc networks," *Ad Hoc Netw.*, vol. 7, no. 8, pp. 1526–1535, 2009.

[10] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," Cryptology ePrint Archive: Rep. 2010/351, 2010.

[11] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proc. Eurocrypt*, 2005, pp. 457–473.

[12] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. ACM Conf. Comput. Commun. Security*, 2006, pp. 89–98.

[13] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attributebased encryption," in *Proc. IEEE Symp. Security Privacy*, 2007, pp. 321–334.

[14] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. ACM Conf. Comput. Commun. Security*, 2007, pp. 195–203.

[15] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute based data sharing with attribute revocation," in *Proc. ASIACCS*, 2010, pp. 261–270.