Abstract

In this project, cryptography is implemented using pairwise key generation technique. When a node wants to communicate with some other node in the network without sharing the information with any other node in the network, it will first check whether the node is in its range or not . If they are not in range, they can communicate implicitly i.e. via a node in between them. They both will then find a private key between them. Using this key, they will be able to share the personal information or data. After receiving the message, the key will be destroyed. In cased of a sybill attack a malicious node attack the whole wireless sensor network and Blom's Scheme will help up to detect and destroy any malicious node in the network . In this project, we can take the number of nodes as per our choice.

INTRODUCTION

1 Introduction

In the recent years there has been an emminent effort to enhance the security in Wireless Sensor Networks in defence & other autonomous organizations.

A Wireless Sensor Network is a collection of sensor nodes which are wirelessly linked. Key establishment is the technique by which two (or more) entities establish a shared secret key. In Blom’s scheme, a trusted party(which is accoiuntable to whole network) gives each participant a secret key and a public identifier, which enables any two participants to independently create a shared key for communicating.

To achieve security in wireless sensor networks, it is important to be able to encrypt and authenticate messages sent among sensor nodes. Due to resource constraints, achieving such key agreement in wireless sensor networks is non-trivial. Blom’s scheme is a symmetric key exchange protocol used in cryptography. But the concerns with Blom’s scheme are the complexity involved in computation as well as memory usage. In this paper, we propose a new key pre-distribution scheme by modifying Blom’s scheme which reduces the computational complexity as well as memory usage.

1.1 PROBLEM STATEMENT

Develop a system which computes a private key between two nodes in order to communicate in a Wireless Sensor Network using Modified Blom's Scheme and detect if any malicious node enters the system and then destroy it. It is made in such a way that no other node will be able to know the computed key.

1.2 OBJECTIVE AND SCOPE OF THE PROJECT

It offers good features on network operation with low resource constrains, data aggregation, and security . This research focuses on key generation in wireless sensor networks and using the key generation technique to detect and destroy any malicious node in the network. The main goal is on the improvement of security. Other requirements, such as the flexibility and scalability are also considered. The main tasks of this research are as follows:

1) To analyse the requirements of wireless sensor network key generation and existing problems. The main key generation schemes are assessed on security, efficiency and operation requirements. The features of each scheme are outlined.

2) To propose a modified key generation scheme based on existing schemes. It includes key generation, key usage and key destroy. The proposed scheme should satisfy many requirements of key generation as possible . Using the technique detection of any malicious node in the network and destruction of such node.

3) To analyse and evaluate the proposed key generation scheme on security strength and performance. Simulation results are required to back the analysis.

1.3 ORGANISATION

Chapter 1 highlights the important aspects of the Wireless Sensor Networks which are related to the Blom's Scheme. In this chapter, details of the Sybil attack and a mechanism to achieve security in wireless sensor networks is discussed. The main focus on modified Blom's Scheme is done so that proposed work can be further continued.

Chapter 2 contains the detailed literature overview of the reviewed research papers, books, journals, and conferences. In this chapter, summarized version of the research papers on Blom's scheme of Pairwise Key Distribution Technique to prevent Sybil in Wireless Sensor Networks are presented.

Chapter 3 covers the System Analysis which explains important characteristics of the project with the help of UML diagrams like data flow diagram, class diagram, and sequence diagram

Chapter 4 is the Performance Analysis which contains the screenshots of the developed application. This chapter also lists the resource requirements needed for the application to run.

Chapter 5 ends with the detailed conclusion and scope of the future work which will be used as a guiding tool for other research scholars to enhance the current work with higher efficiency.

LITERATURE SURVEY

The implementation of this project requires an extensive knowledge of-

1) Wireless Sensor Networks
2) Sybil Attack
3) Cryptography
4) Blom's Scheme
5) Modified Blom's Scheme

2.1 Wireless Sensor Networks

Wireless sensor networks (**WSN**), aka **wireless sensor and actuator networks** (**WSAN**), are spatially distributed autarchic sensors to *oversee* physical or environmental conditions, such as temperature, sound, pressure, etc. and to collaboratively pass statistics through the network to the main location.

The evolution of wireless sensor networks was motivated by defence applications such as battlefield monitoring; today these networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

The WSN is a collection of "nodes" – ranging from a few to several hundred or even thousand, where each node is inter-connected to other sensors. Each such sensor network node typically consists of three parts : a radio transceiver with an internal antenna or connection to an external antenna , an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting.

2.1.1 Applications of WSN's

1) Area monitoring

Area monitoring is the most common application of WSNs. In area monitoring, the Wireless Sensor Network is deployed over a stretch of land/region where some activity is to be recorded. Armed forces inspection is an example of use of sensors detects enemy invasion; a civilian example is the geo-fencing of gas or oil pipelines.

2) Health care monitoring

The medical applications of Wireless Sensor Network can be of two types wich includes wearable type and the implanted type . Wearable devices are used on the body surface of a human or just at near the user. The implantable medical devices are the devices which are embedded into the human body. There are many other applications of the WSN e.g. body position measurement and determining the location of the person, overall monitoring of ill and unfit patients in hospitals , penitentiaries and homes. Body area networks can collect information about an individual's health, fitness, and energy expenditure.

3) Air pollution monitoring

Wireless sensor networks have been used in a number of sites to monitor the concentration(ppm) of dangerous gasses and substances for citizens. Ad hoc wireless links have an advantage rather than wired installations, which make them more mobile and provide ease for testing readings in different areas.

4) Forest fire detection

A network of Sensor Nodes installed in a forest can be used to detect whether a fire has started. The nodes equipped with sensors can be used to measure temperature-change, humidity, and concentration and nature of gasses which are being produced during a forest-fire. Thanks to Wireless Sensor Networks, the early detection is very important firefighters to take any action, the fire brigade will be able to know about the forest fire how it is generated and what is the area affected.

5) Landslide detection

A landslide detection system can make use of a WSN to detect the movements of soil-form and changes in various environmental parameters that may occur before or during a landslide. The data gathered using the WSN can be used to detect the occurance of landslide before happening.

6) Water quality monitoring

Water quality monitoring involves analyzing properties water properties in various water bodies. Use of many wirelessly distributed sensors enables the creation of a more accurate map of the water status, and allows the permanent deployment of monitoring stations in locations with difficult access, without the manual retrieval of data.

7) Natural disaster prevention

Wireless sensor networks can be used to prevent many of the natural disasters, like drought , floods , femine and earthquake by predetermining the data from the environment and using it to monitor association of one activity with other.

8) Machine health monitoring

WSN have been developed for machinery and condition-based maintenance (CBM) .They offer significant cost reduction and enable new functionalities to the system.

Wireless sensors can be placed in remote locations which are difficult or impossible to reach with a wired system.

9) Data center monitoring

Due to the high number of servers in a data center, cabling and IP Address generation is a big issue . To overcome that problem of wiring , more and more servers are fitted out with wireless sensors to sense the temperature, so as to monitor the temperatures of racks of servers. As per ASHRAE recommendation , up to 6 temperature sensors per rack of server, gives an advantage compared to traditional cable sensors.

10) Data logging

WSN are also used for the data collection and monitoring of environmental information, this can be as simple as the monitoring of the temperature in a room to the level of water in tanks in power plants. The statistical information can then be used to show how systems have been working. The main advantage of WSNs over conventional system is the live data that is being recorded.

11) Water/wastewater monitoring

Monitoring the quality and level of water includes many activities such as checking the quality of underground or surface water and ensuring a country’s water infrastructure for the benefit of both human and animal. It may be used to protect the wastage of water.

.

2.2 Sybil Attack

The Sybil attack is an attack in which the reputation system is affected by creating multiple identities in Wireless Sensor Network . It is named after the subject of the book Sybil,which is a case study of a woman diagnosed with dissociative identity disorder. The name was suggested in or before 2002 by Brian Zill at Microsoft Research.

In a Sybil attack, the attacker penetrate the system of a WSN by creating a large number of pseudo anonymous identities, using them to gain a disproportionately large influence. A system's vulnerability to be attacked by a Sybil attack depends on the technique using which identities can be generated, the degree to which the system accepts inputs from entities that do not have a chain of trust linking them to a trusted entity, and whether the reputation system treats all entities identically.

An entity on a P2P network is a piece of interacing software having access to local resources. An entity is made available to all other entities of the network using a particular identity . More than one identity can correspond to a single entity. In other words, the mapping of identities to the entities in the WSN is many to one. Entities in P2P networks use multiple identities for purposes of resource sharing, reliability, and integrity. In P2P networks , the identity can be used as an abstraction so that a remote entity can be aware of identities without necessarily knowing the correspondence of identities to local entities. By default, each distinct identity is usually assumed to correspond to a distinct local entity. In reality, many identities may correspond to the same local entity.

A faulty node or an adversary may present multiple identities to a peer-to-peer network in order to appear and function as multiple distinct nodes. After becoming part of the peer-to-peer network, the adversary may then overhear communications or act maliciously. By masquerading and presenting multiple identities, the adversary may be able to affect voting outcomes or even substantially control the network.

In the context of (human) online communities, such multiple identities are sometimes known as sockpuppets.

2.2.1 Example of Sybil Attack

A notable Sybil attack (in conjunction with a traffic confirmation attack) was launched against the Tor anonymity network for several months in 2014 by unknown perpetrators. Many in the network security community suspect the NSA/CIA to be responsible for the attack, and some speculate that the attack may have been connected to the investigation into the Silk Road website.

2.2.2 Prevention of Sybil Attack

Validation techniques can be used to prevent Sybil attacks and dismiss masquerading hostile entities. A local entity may accept a remote identity based on a central authority which ensures a one-to-one correspondence between an identity and an entity and may even provide a reverse lookup. An identity may be validated either directly or indirectly. Indirect validation, the local entity queries the central authority to validate the remote identities. In indirect validation, the local entity relies on already accepted identities which in turn vouch for the validity of the remote identity in question.

Identity-based validation techniques generally provide accountability at the expense of anonymity, which can be an undesirable trade-off especially in online forums that wish to permit censorship-free information exchange and open discussion of sensitive topics. A validation authority can attempt to preserve users' anonymity by refusing to perform reverse lookups, but this approach makes the validation authority a prime target for attack. Alternatively, the authority can use some mechanism other than knowledge of a user's real identity - such as verification of an *unidentified* person's physical presence at a particular place and time - to enforce a one-to-one correspondence between online identities and real-world users.

Sybil prevention techniques based on the connectivity characteristics of social graphs can also limit the extent of damage that can be caused by a given Sybil attacker while preserving anonymity, though these techniques cannot prevent Sybil attacks entirely, and may be vulnerable to widespread small-scale Sybil attacks. Examples of such prevention techniques are SybilGuard and the Advogato Trust Metric and also the sparsity-based metric to identify Sybil clusters in a distributed P2P based reputation system.

2.3 Cryptography

Crypto is derived from the Greek word '*kryptós'* which means "hidden" or "secret"; and graphy is derived from the Greek word '*graphein'* which means "writing". Cryptography is the practice and study of techniques for secure communication in the presence of third parties called adversaries. More generally, cryptography is about constructing and analyzing protocols that prevent third parties or the public from reading private messages; various aspects of information security such as data confidentiality, data integrity, and authentication, are central to modern cryptography. Modern cryptography exists at the intersection of the disciplines of mathematics, computer science, and electrical engineering. Applications of cryptography include ATM cards, computer passwords, and electronic commerce.

Cryptography prior to the modern age was effectively synonymous with *encryption,* the conversion of information from a readable state to apparent nonsense. Modern cryptography is heavily based on mathematical theory and computer science practice; cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in practice by any adversary. It is theoretically possible to break such a system, but it is infeasible to do so by any known practical means.

2.3.1 Why Cryptography is necessary for a Distributed System

Supporting the facilities of a distributed system, such as resource distribution, requires the use of an underlying message passing system. Such systems are, in turn, reliant on the use of a physical transmission network, upon which the messages may physically be communicated between hosts.

Physical networks and, therefore, the basic message passing systems built over them are vulnerable to attack. For example, hosts may easily attach to the network and listen in on the messages (or 'conversations') being held. If the transmissions are in a readily understandable form, the eavesdroppers may be able to pick out units of information, in effect stealing their information content.

Aside from the theft of user data, which may be in itself of great value, there may also be system information being passed around as messages. Eavesdroppers from both inside and outside the system may attempt to steal this system information as a means of either breaching internal access constraints or to aid in the attack of other parts of the system. Two possibly worse scenarios may exist where the attacking system may modify or insert fake transmissions on the network. Accepting faked or modified messages as valid could lead a system into chaos.

Without adequate protection techniques, Distributed Systems are extremely vulnerable to the standard types of attack outlined above. The encryption techniques discussed below aim to provide the missing protection by transforming a message into a form where if it were intercepted in transit, the contents of the original message could not be explicitly discovered. Such encrypted messages, when they reach their intended recipients, however, are capable of being transformed back into the original message.

There are two main frameworks in which this goal may be achieved, they are named Secret Key Encryption Systems and Public Key Encryption Systems.

2.3.2 Types of key encryption

1. Public Key Encryption

Asymmetric encryption, also known as public-key encryption utilizes a pair of keys – a public key and a private key. If you encrypt data with the public key, only the holder of the corresponding private key can decrypt the data, hence ensuring confidentiality.

Many “secure” online transaction systems rely on asymmetric encryption to establish a secure channel. SSL, for example, is a protocol that utilizes asymmetric encryption to provide communication security on the Internet.

Asymmetric encryption algorithms typically involve exponential operations, they are not lightweight in terms of performance. For that reason, asymmetric algorithms are often used to secure key exchanges rather than used for bulk data encryption.

2. Private Key Encryption

Symmetric encryption, as the name suggests, means that the encryption and decryption operations utilize the same key. For two communicating parties using symmetric encryption for secure communication, the key represents a shared secret between the two.

Symmetric encryption is typically more efficient than asymmetric encryption and is often used for bulk data encryption.

2.4 Blom’s Scheme

Blom’s scheme is a symmetric threshold key exchange cryptography protocol. It allows any pair of users in the system to find a unique shared key for secure communication. In this scheme, a network with N users and a collision of fewer than t+1 users cannot reveal the keys which are held by other users. Thus, the security of the network depends on the chosen value of t, which is called Blom’s secure parameter (t<<N). A Larger value of t leads to greater resilience but a very high value increases the amount of memory required to store key information.

Generation of Public matrix: Initially, a central authority or base station first constructs a (t + 1) ×N matrix P over a finite field GF (q), where N is the size of the network and q is the prime number. P is known to all users and it can be constructed using a Vandermonde matrix. It can be shown that any t+1 columns of P are linearly independent when i=1, 2,…N are all distinct.

$$P = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ n_1 & n_2 & n_3 & \cdots & n_N \\ {n_1}^2 & {n_2}^2 & {n_3}^2 & \cdots & {n_N}^2 \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ {n_1}^t & {n_2}^t & {n_3}^t & \cdots & {n_N}^t \end{bmatrix}$$

Figure 1: Vandermode matrix

Generation of Secret Key (Private matrix): The central authority or the base station selects a random (t + 1) × (t + 1) symmetric matrix S over GF (q), where S is secret and only known by the central authority. AN × (t + 1) matrix A = $(S. P)^T$ is computed which is needed for generating the shared key. K =A. P = $(S. P)^T. P = P^T. S^T.P = P^T.S.P = (A.P)^T = K^T$

Generation of shared key by the user pair: User pair (i, j) will use K_{ij}, the element in row I and column j in K, as the shared key. Because K_{ij} is calculated by the ith row of A and the jth column of P, the central authority assigns the ith row of A matrix and the ith column of

P matrix to each user i, for 1, 2….. N. Therefore, when user i and user j need to establish a shared key between them, they first exchange their columns of P, and then they can compute K_{ij} and K_{ji}, respectively, using their private rows of A. It has been proved in that the above scheme is t-secure if any t + 1 columns of G are linearly independent. The t-secure parameter guarantees that no compromise of up to t nodes has any information about K_{ij} or K_{ji}.

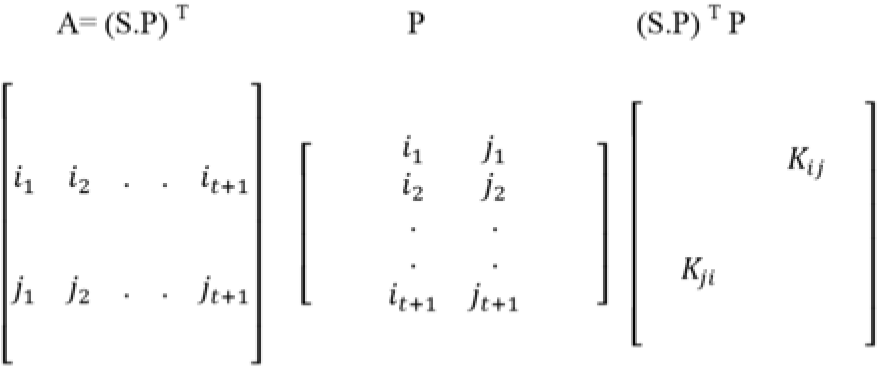


Figure 2: Generating keys in Blom’s Scheme

2.4.1 Example of Blom’s Scheme

The example below shows the working of proposed Scheme that uses a random matrix as a public key and generates a private key. Let us consider a network with 4 nodes and the following parameters:

1. Let C be the Central Authority or Base Station.

2. Secure parameter t = 3, which says if more than 3 nodes in the network are compromised, it is not possible to find the keys of other users.

3. Prime number q= 31.

4. As the public matrix (P) should be of the order (t+1) * (t+1), P can be taken as any random (3+1) * (3+1) matrix.

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 3 & 1 \\ 4 & 0 & 9 & 5 \end{bmatrix}$$

5. The secret symmetric matrix can be obtained as follows:
   Let us take a random matrix,

$$A=\begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 2 & 3 & 1 \end{bmatrix} \qquad B=\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 2 \\ 1 & 0 & 1 & 3 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

We obtain the secret matrix S by taking the product of two matrices A and B,

$$S=\begin{bmatrix}3 & 2 & 2 & 4\\ 2 & 6 & 1 & 5\\ 2 & 1 & 2 & 4\\ 4 & 5 & 4 & 14\end{bmatrix}$$

6. Now, we calculate matrix A using,
   $A = (S.P)^T \bmod q$

$$(S.P)=\begin{bmatrix}25 & 8 & 53 & 36\\ 30 & 5 & 60 & 40\\ 23 & 6 & 49 & 31\\ 73 & 12 & 155 & 95\end{bmatrix}$$

$$A= (S.P)^{T} \bmod 31 = \begin{bmatrix}25 & 8 & 22 & 5\\ 30 & 5 & 29 & 9\\ 23 & 6 & 18 & 0\\ 11 & 12 & 0 & 2\end{bmatrix}$$

7. Once A is calculated, each sensor node memory is filled with unique row chosen from A with the corresponding index. These are the private keys for the nodes.

$K_{A,B} =$

$$\begin{bmatrix}8 & 5 & 6 & 12\end{bmatrix}\begin{bmatrix}3\\ 1\\ 3\\ 9\end{bmatrix}$$

$K_{B,A} =$

$$\begin{bmatrix}22 & 29 & 18 & 0\end{bmatrix}\begin{bmatrix}2\\ 0\\ 1\\ 0\end{bmatrix}$$

As seen above, both the nodes generate a common key and further communication can be done using the shared key generated.

2.4.2 Disadvantages of Blom's Scheme

(1) Any two sensors can definitely establish a pairwise key when there are no compromised sensors;

(2) Even with some nodes compromised, the others in the network can still establish pairwise keys;

(3) A node can see the usual keys to specify whether or not it can establish a pairwise key and thereby help reduce communication overhead.

2.5 Modified Blom's Scheme

There are two ways to implement Modified Blom's Scheme

Method 1:

In this method, we make use of the Bloms scheme. As stated earlier the Blom's scheme makes use of Vandermonde matrix which is a public matrix and this matrix is responsible for all computations in generating keys and since it's a public matrix, it could be known even to the eavesdroppers. We choose all the values or elements of this matrix to be distinct so that any $\lambda + 1$ columns of G are linearly independent i.e., to generate unique keys. But when the value of λ increases to larger values, then the number of rows of the public matrix increase and this, in turn, results in a greater value in the columns because the column values increase in a geometric series. We know in wireless sensor networks, sensor nodes contains limited memory and energy, wherein Blom's scheme for any two nodes to generate a common key, each node should store column of public matrix and row of the calculated secret matrix and this would be difficult for sensor networks to store both the row and column in the memory for a large network.

To reduce the computation and memory overhead in Blom’s scheme, instead of using Vandermonde matrix [12] we propose the use an Adjacency Matrix as the public matrix. As the Adjacency matrix is a square matrix with 1s and -0s, it reduces the complexity of calculating values for all the elements corresponding to the columns in Vandermonde matrix. This adjacency matrix is formed in such a way that all nodes that are neighbors of a particular node are filled with 1s and remaining with q-1(since public matrix cannot contain 0s).

Another advantage of using Adjacency matrix is it reduces the cost of saving the columns in the memory of sensor because any node can easily generate Adjacency matrix of known size. Similar to Blom’s scheme the operation which are to be performed to generate the keys will depend on the prime number i.e., the number which depends on the desired key length.

$$\begin{vmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{vmatrix}$$

Figure 3: Adjacency Matrix

$$\begin{vmatrix} 1 & 1 & 28 & 1 \\ 1 & 28 & 28 & 28 \\ 28 & 28 & 1 & 1 \\ 1 & 28 & 1 & 28 \end{vmatrix}$$

Figure 4: Changed modified Adjacency Matrix for Modulo 29

The original binary form Adjacency Matrix is depicted in Figure 3. The only change that is made to the Adjacency Matrix is all zeros are replaced with the q-1(prime number -1). We can see from Figure 4 that the Adjacency Matrix consists of only two different numbers one’s and q- 1(prime number -1), hence all the further calculations will be very simple. Key generation technique is similar to that used in the Blom’s scheme [7]. Following are the steps involved in calculating the key.

Step 1: Generating a G matrix. We first select a primitive element from a finite field GF(q), where q is larger than the desired key length (and also q > N), and then construct adjacency matrix G of size N ×N as discussed in the previous section depending on the node neighbours. Then depending on the λ value first λ rows along with N columns are selected as the public matrix. Let G (j) represent the jth column of G, in Blom's scheme G (j) is provided to node j but in our case, each node knows its neighbors and therefore the memory usage for storing G (j) at a node is not required.

Step 2: Generating key spaces. The Central Authority generates ω random, symmetric matrices D1, . . . , Dω of size (λ+1)×(λ+1). We then compute the matrix Ai= (Di. G) T. Let Ai(j) represent the jth row of Ai.

Step 3: Computing Secret Key. The central authority stores each row of the matrix A in the node memory with the corresponding index. Now if node I want to communicate with node j then node I multiply the row Ai with column Gj and the result will be the secret key.

Method 2:

Blom's scheme is a prominent key management scheme but its shortcomings include large computation overhead and memory cost. We propose a new scheme in this project that modifies Blom's scheme in a manner that reduces memory and computation costs.

Let p be a large prime number which is less than the number of nodes in the network. Each n user U in the network is assigned a distinct prime number Ru (mod p). Then the main authority chooses three random numbers a(mod p), b(mod p) and c(mod p). For each user U, the main authority calculates the numbers A_U = a+b.r_U (mod p) and b_U = b+c.r_U (mod p). Each user U forms the linear polynomial g_U (x) = Au + b_U.x(mod p) . Now, if A wants to communicate with B, then A & B computes K_{AB} = G_A (R_B) and K_{BA} = G_B (R_A) respectively.

Let us understand this with an example

- Consider a network consisting of two users A & B.
- Let p=23 , r_A = 11 , r_B = 3 .

- The main authority chooses three random numbers a=8, b=3, and c=1.
- Then, $a_A = 8 + 11*3 \pmod{23} = 18$ and $b_A = 3 + 11*1 \pmod{23} = 14$;
- $a_B = 8 + 3*3 \pmod{23}=17$ and $b_B = 3 + 1*3 \pmod{23} = 6$.
- Hence, $g_A(x) = 18 + 14x$ and $g_B(x) = 17 +6x$.
- Now, $K_{AB} = g_A(r_B) \bmod 8 = 14$ and $K_{BA} = g_B(r_A) \bmod 8 = 14$.
- Therefore $K_{AB} = K_{BA}$. K_{AB} is the private key between A and B.

3.1.1 DFD LEVEL 0

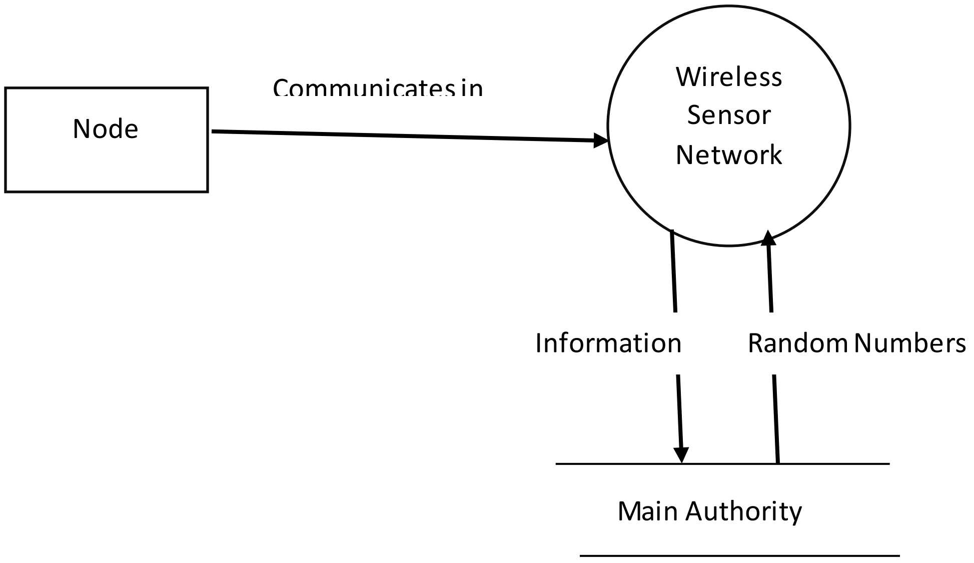


Figure 3: DFD Level 0

3.1.2 DFD LEVEL 1

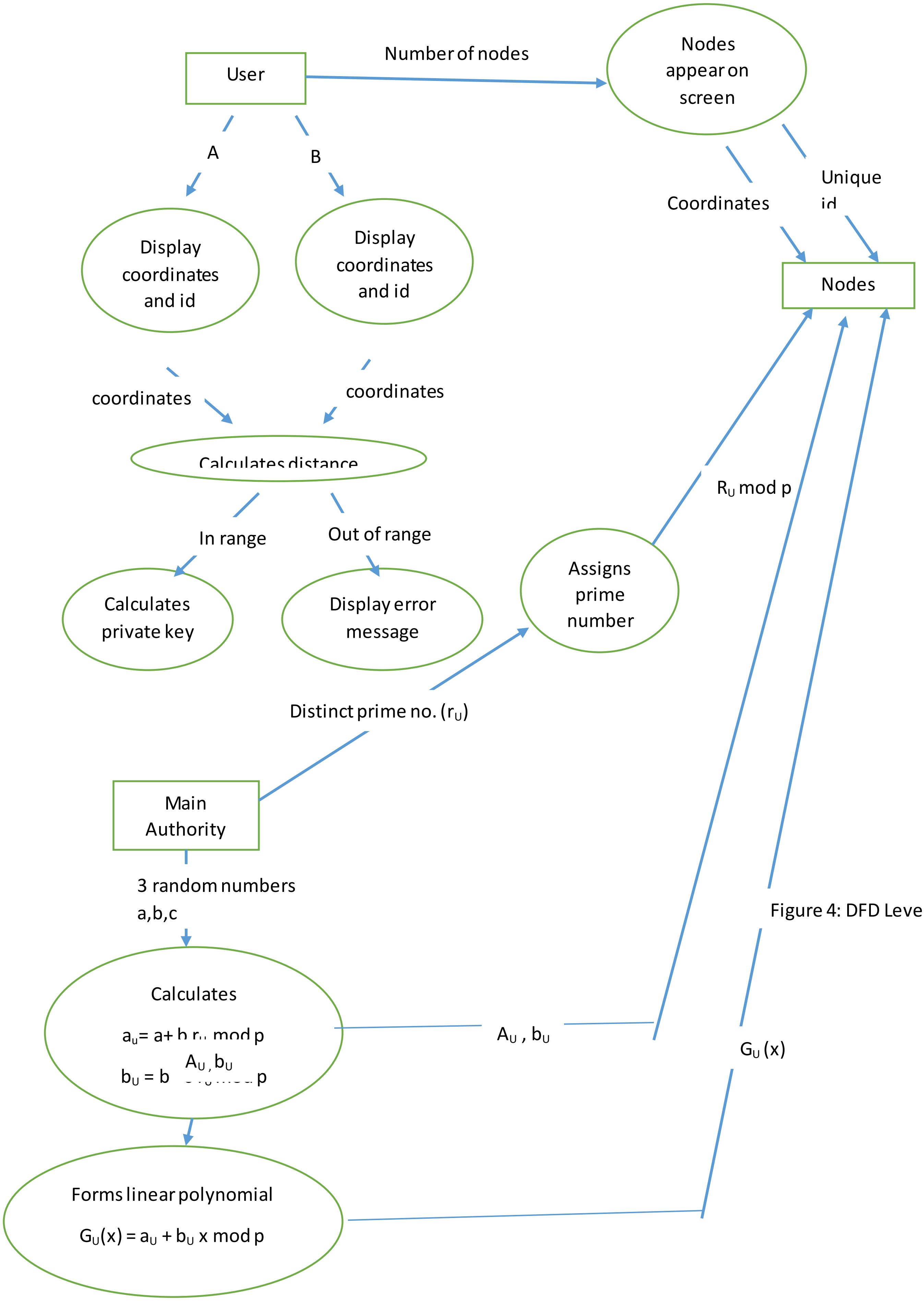


Figure 4: DFD Level 1

3.2 Class Diagram

Class diagrams are visual representations of the static structure and composition of a particular system using the conventions set by the Unified Modelling Language (UML). Out of all the UML diagram types, it is one of the most used ones. System designers use class diagrams as a way of simplifying how objects in a system interact with each other. Using class diagrams, it is easier to describe all the classes, packages, and interfaces that constitute a system and how these components are interrelated.

Classes in class diagrams are represented by boxes that are partitioned into three:

1. The top partition contains the name of the class.
2. The middle part contains the class's attributes.
3. The bottom partition shows the possible operations that are associated with the class.

Class Diagram

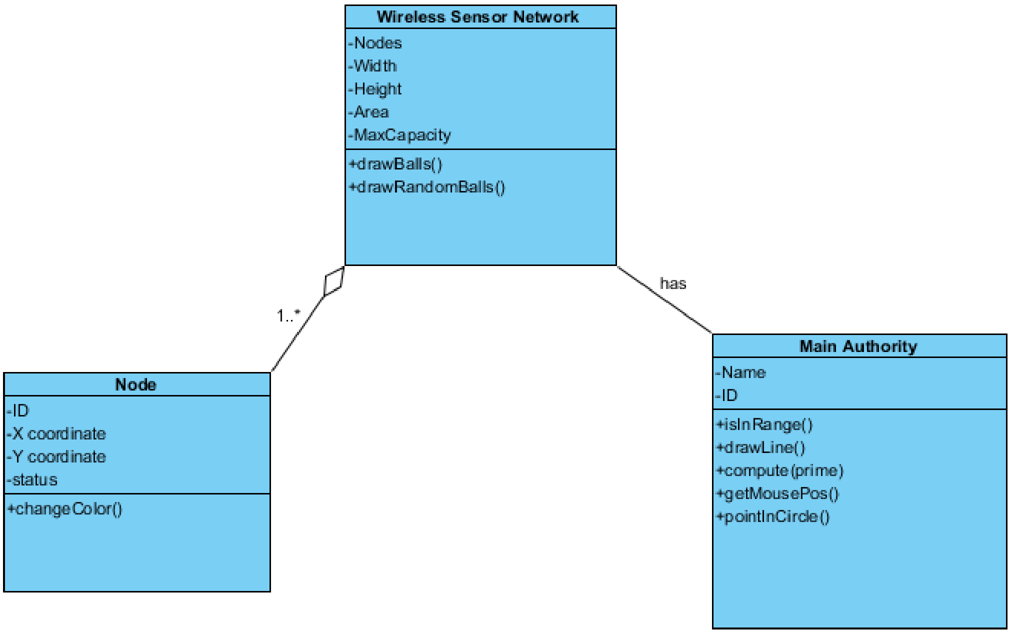


Figure 5: Class Diagram

Use Case Diagram

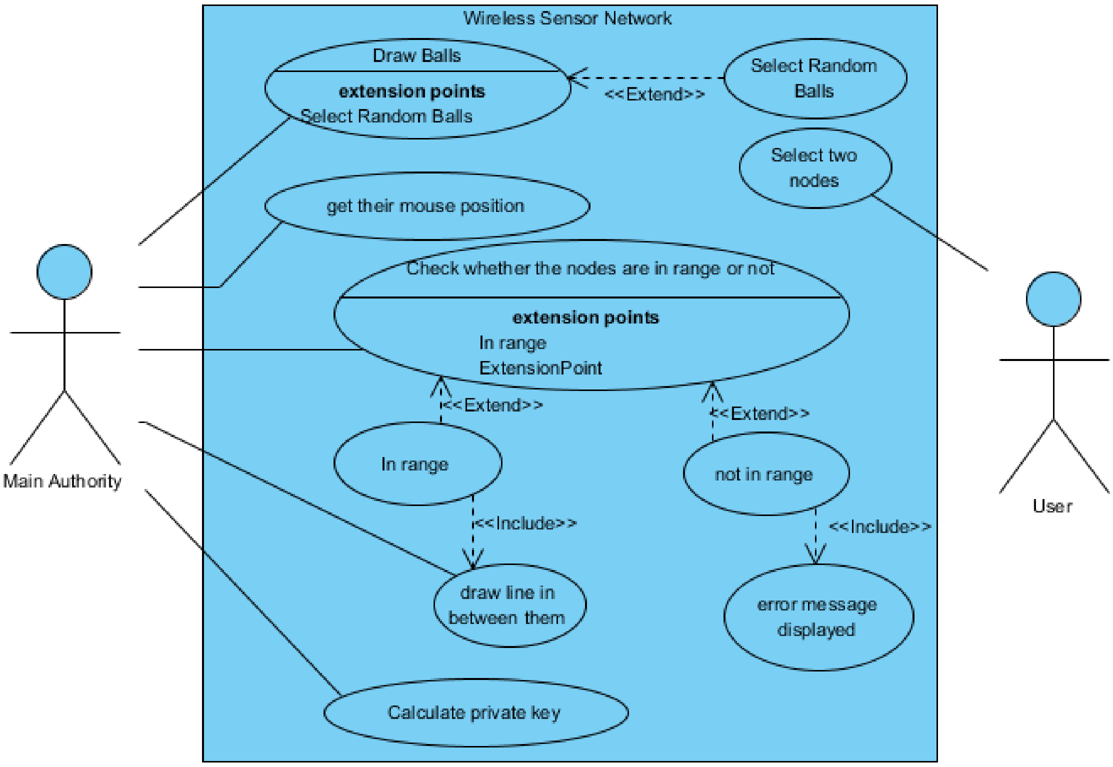


Figure 6: Use Case Diagram

Sequence Diagram

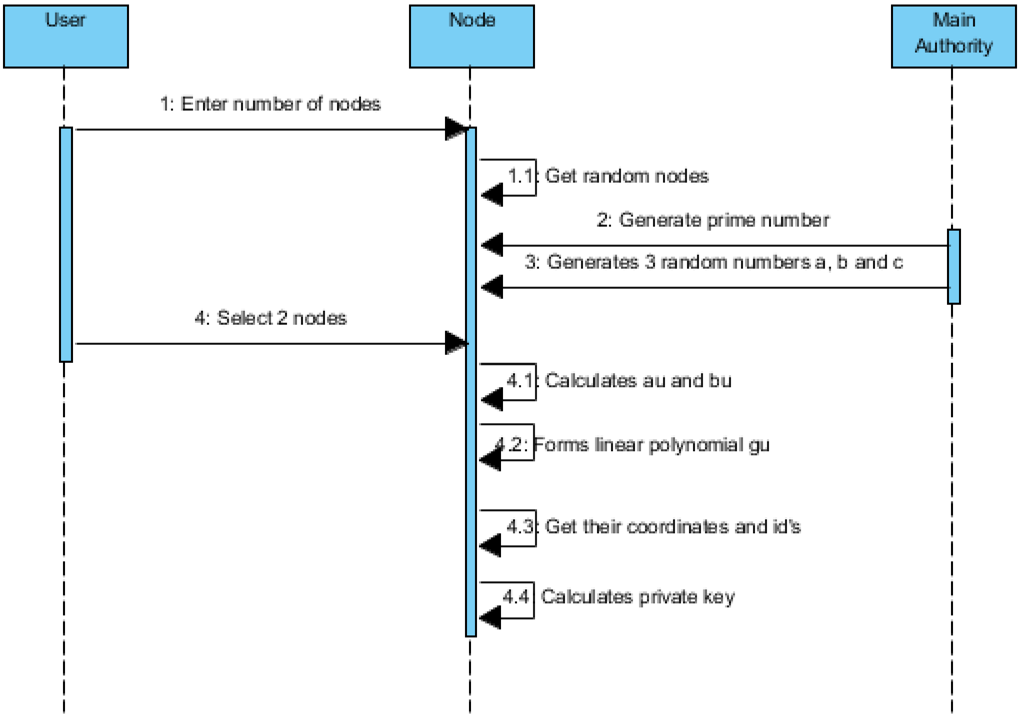

User
Node
Main Authority
1: Enter number of nodes
1.1: Get random nodes
2: Generate prime number
3: Generates 3 random numbers a, b and c
4: Select 2 nodes
4.1: Calculates au and bu
4.2: Forms linear polynomial gu
4.3: Get their coordinates and id's
4.4 Calculates private key

Figure 7: Sequence Diagram

3.5 ALGORITHM

1. Each of n users is given initial secret keying material and public data.

2. Each pair of users UA, U_B may compute the secret key K_{AB}=K_{BA}

3. Let p be a large p≤n. everyone has knowledge of the prime p.

4. Each n user U in the network is assigned a distinct public number r_{un}(mod p).

5. Main authority chooses three secret random numbers a, b, and c(mod p).

6. For each user, U. Trent calculates the numbers

$$a_u = a+b.r_u(\text{mod } p) \qquad b_u = b+c.r_u(\text{mod } p)$$

And sends them via a secure channel

7. Each user U forms the linear polynomial

$$g_u(x) = a_u + b_u.x(\text{mod } p).$$

8. If A wants to communicate with B, then A computes

$K_{AB} = g_A(r_B)$, while B computes $K_{AB} = g_B(r_A)$, which is the private encryption key.

3.6 SIMULATION ENVIRONMENT

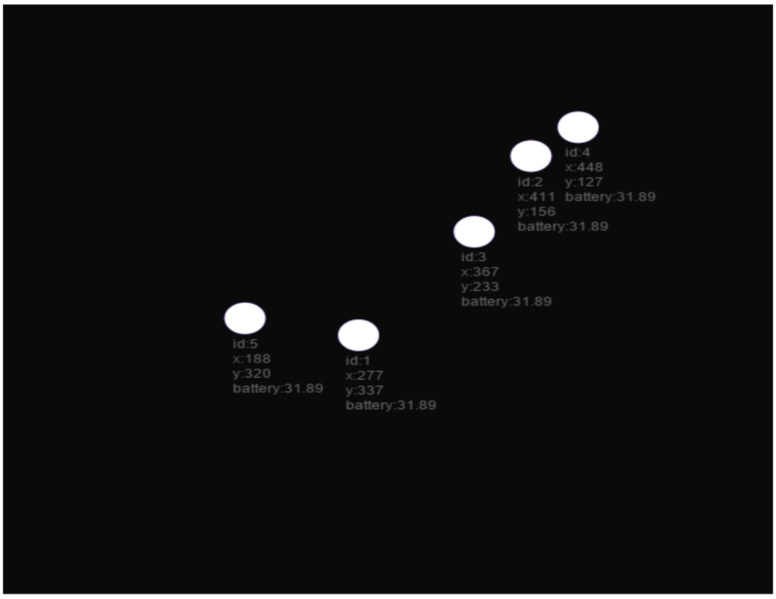
Summary of the Simulation Environment	
Simulator	JavaScript
Simulation Area	640 pixels * 480 pixels
Allocation	Dynamic
Transmission Range of Legitimate node	200 pixels
Number of nodes	2 nodes – 80 nodes
An observation period	Variable, by selecting different number of nodes
Communication Direction	0-2π
Number of nodes communicating simultaneously	2
Presentation of nodes	Simultaneous
Framework	HTML Canvas(p5.js)

Table 1: Simulation Environment

PERFORMANCE ANALYSIS

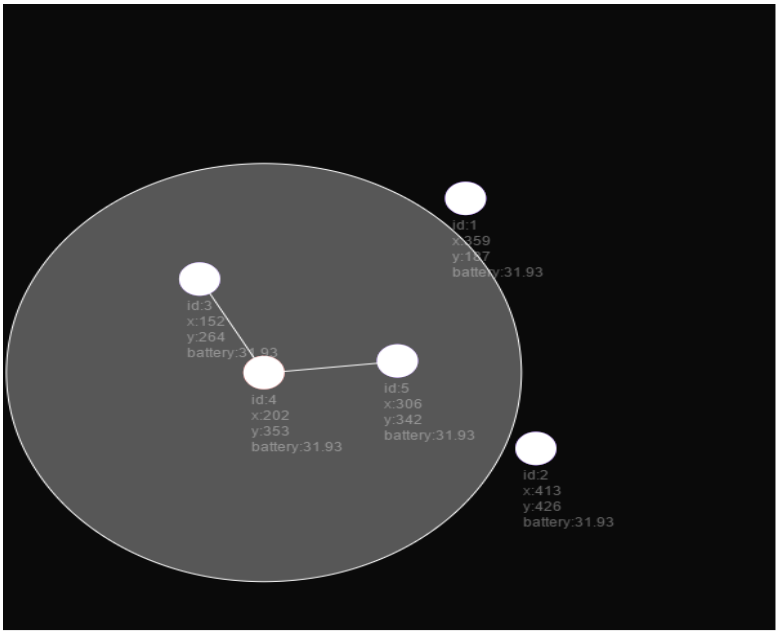
4.1 SCREENSHOTS

1. Run index.html

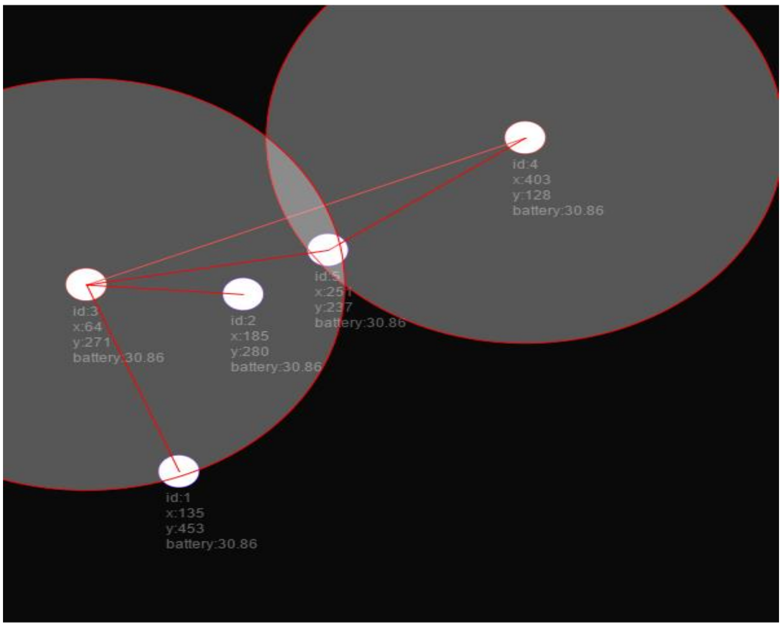


Case 1: When nodes are not in range

1. Select the first node

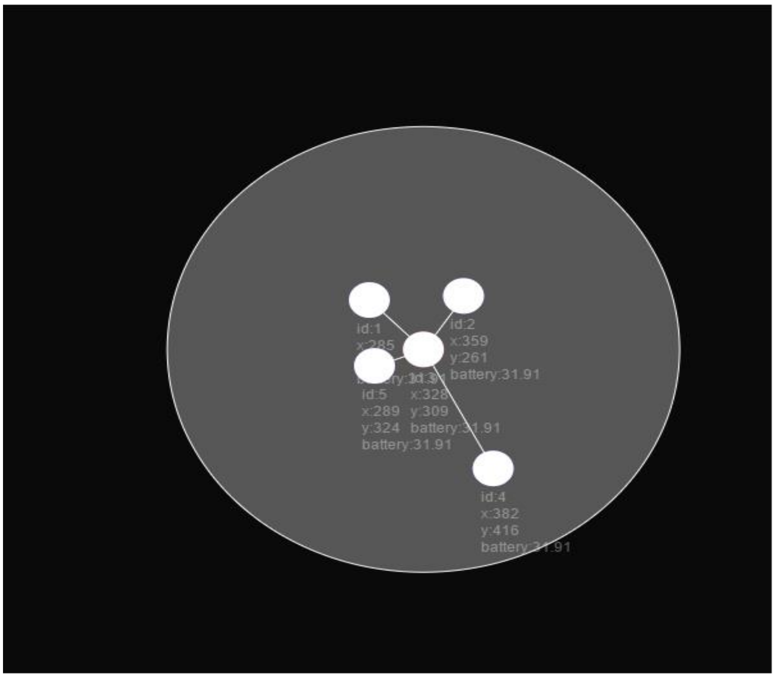


2. Select the second node

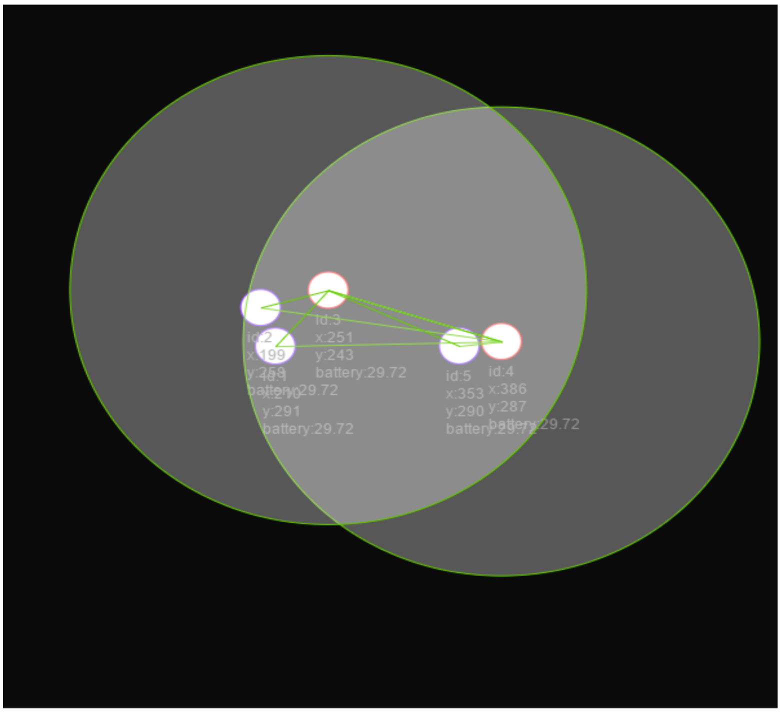


Case 2: When nodes are in range

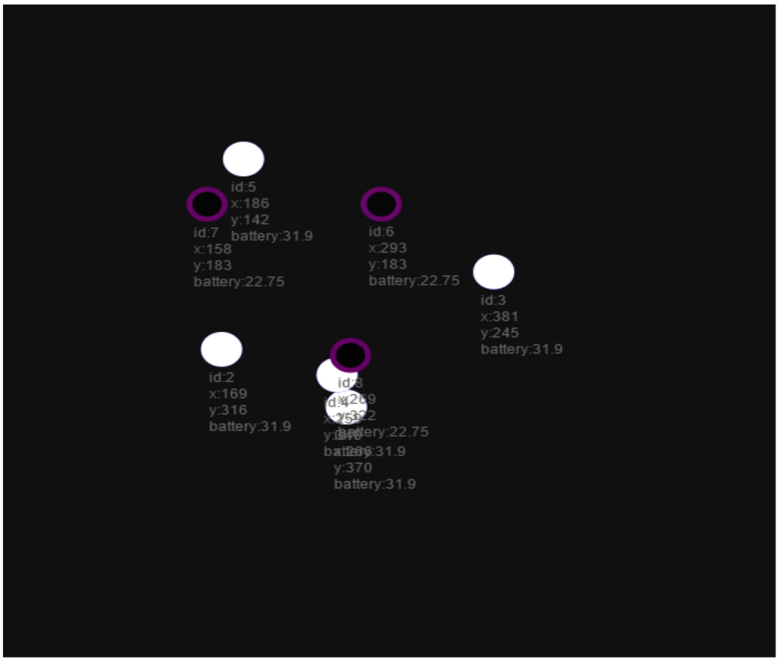
1. Select the first node



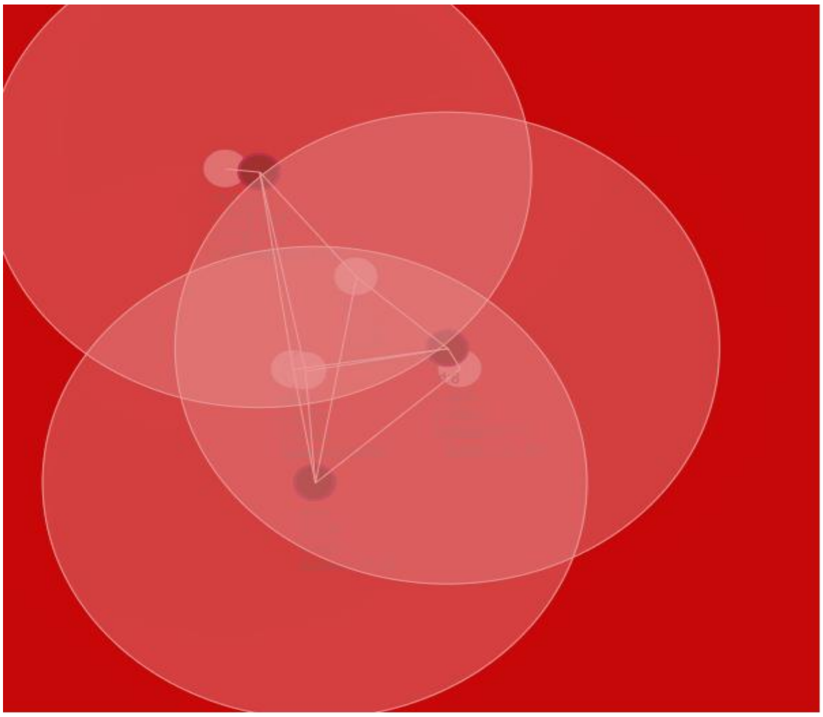
2. Select the second node



2.Intoduce malicious node on network

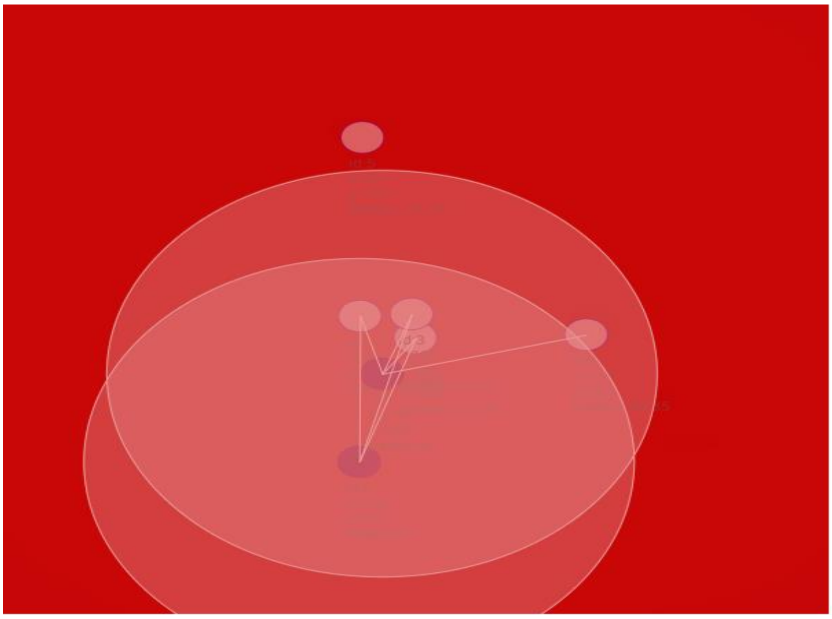


2.1 Malicious node try to establish link with other neighbouring nodes & in return legitimate node identifies malicious node using bloms's scheme

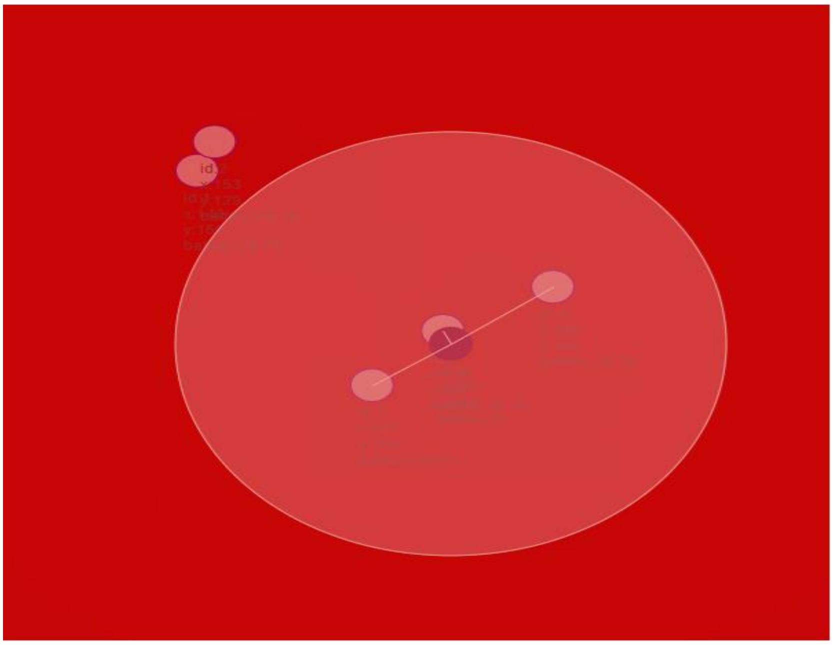


2.2 Legitimate node tries to destroy malicious nodes using the signal strength

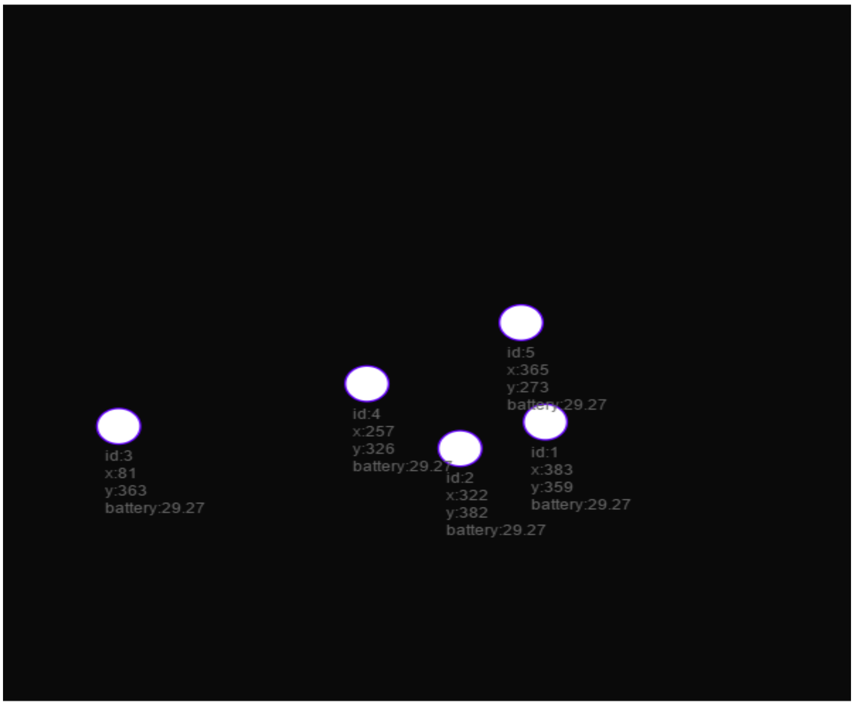
1st malicious node destroyed



2nd malicious node gets destroyed



3rd Malicious node gets destroyed and legitimate nodes broadcast system is secure


id:5
x:365
y:273
battery:29.27
id:4
x:257
y:326
battery:29.27
id:3
x:81
y:363
battery:29.27
id:2
x:322
y:382
battery:29.27
id:1
x:383
y:359
battery:29.27