Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1011407, 14 pages
https://doi.org/10.1155/2022/1011407

Research Article

Smart Communication and Security by Key Distribution in Multicast Environment

Manisha Yadav,[1] **Karan Singh**,[2] **Ajay Shekhar Pandey**,[3] **Adesh Kumar**,[4] **and Rajeev Kumar**[5]

[1]*Department of Electronics and Communication Engineering, IET, Dr. Rammanohar Lohia Avadh University, Ayodhya, India*
[2]*School of Computer and System Sciences, Jawaharlal Nehru University, New Delhi, India*
[3]*Department of Electrical Engineering, Kamla Nehru Institute of Technology (KNIT), Sultanpur, India*
[4]*Department of Electrical & Electronics Engineering, School of Engineering, University of Petroleum and Energy Studies, Dehradun, India*
[5]*Department of Computer Science & Engineering, IET, Dr. Rammanohar Lohia Avadh University, Ayodhya, India*

Correspondence should be addressed to Karan Singh; karan@mail.jnu.ac.in

Received 12 November 2021; Revised 3 February 2022; Accepted 26 February 2022; Published 19 March 2022

Academic Editor: Danfeng Hong

The service providers are aiming to provide multicast applications, primarily in the area of content delivery and secure wireless networks, due to the increased adoption of network systems and demand for secured wireless networks communication. Cryptography enables the users to send information across insecure networks using data encryption and decryption with key management. The research paper proposes a unique way of safeguarding network systems using cryptographic keys, as well as a fuzzy-based technique for improving security by reducing symmetric and asymmetric key overhead. To enable efficient communication, fuzzy-based rules with security triads and cryptographic key management methods are used. When the key distribution is decentralized, security implementation becomes more difficult, and multiple types of attacks are possible. Fuzzy logic-based key management methods are used in addition to offering a novel technique for secure cryptography systems. The novelty of the work is that the simulation work is also carried out to verify the data in on-demand distance vector (AODV) multicast wireless routing that supports 100 nodes with network performance parameters such as delay, control overhead, throughput, and packet delivery ratio. The system supports up to 128-bit key embedded with 128-bit plain data in cryptographic encryption and decryption.

1. Introduction

The safety and security of communication have become now the most important aspect because of the requirement of privacy and authentication in wireless multicast communications [1]. Group communication is based on broadcast or multicast technologies, such as internet protocol multicast that offers efficient transmission of group messages using encryption, signatures, authentication, and integrity, comparable to secure two-party communication (STPC) [2]. Cryptographic technologies are used to secure in-group communication. Multicast is a packet transmission method that sends a data packet to a large number of people [3]. A duplicate copy of the package is sent to everyone. Over the last few years, varieties of technologies have emerged that take advantage of new possibilities in the form of a new basic structure for key distribution and key creation in cryptography [4]. In in-group communication, multiple messages transmission is required at the same time to transmit in multicast groups of senders and receivers at reduced bandwidth requirements. The key distribution must ensure to users that various channels are not allowed to unauthorized users and unauthorized access to use the medium. It may access only when the users are fully authorized in the term of security [5]. In in-group communication, there is always a possibility of users being together and separated through the network system anytime means any of them may join the group or may leave the group communication [6]. Nothing like unicast communication, this communication link ends with the disappearance of a group member.

The distribution cycle involved in secure group communication is depicted in Figure 1. To ensure the security of group communication in multicast communication, group communication information should be restricted whenever a member leaves or joins the group. The members will require new keys to do so. Multicasting is a very unique communication technique that facilitates group communications and applications, in which data is sent to a group of users at the same time while maintaining high security and using fewer network resources. As a result, most group-oriented applications, such as software delivery, multiple users video conferencing, and remote learning, are expected to become more practical shortly than previously announced network systems [7].

Secure communication and efficient key management bring the requirements of a cryptographic key management system. A highly protected network system, information, data, and nodes are required for security. Security is the most important feature, which is required in the development of a network system. The network security depends on the key distribution and the policy for the cryptosystem. Network security vulnerabilities emerge from various poor improvement practices, the new method of attacks, and unsecured connections between node-to-node network-based systems. Confidentiality is one of the most vital factors of data and information that are transferred to the node. It is also an important factor of a secure network with the keys distribution concept. The estimation of a secured network has played an important role in transferring, sharing, and accessing information at various nodes. Somehow, key distribution time is also the most appropriate stage to estimate the security of the network because this stage is the first step towards secure communication. This has a positive impact on the overall security, cost, and efforts. Cryptographic systems are also needed to understand how various components of a network interact with each other to secure and enhance the reliability of key distribution during the passing of information [8].

Some experts emphasized the adaptive key management and privacy-protection aggregation scheme with revocation of user data in the smart communication to prevent the appearance of nontrusted nodes. In particular, they examine a light collection scheme to enable aggregate certification first, which protects the nontrusted aggregator from revelations of personal user data. Furthermore, a proposal for an adaptive key management system with efficient repeal, in which users can update their encryption keys automatically if any user is not included or is out of the system. End key time is resolved to stand up to the user to adaptive key management. Security analysis shows that at the same time, forward and backward secrecy is taken under consideration for performance evaluation [9].

The research work proposes a fuzzy rules-based secure and lightweight scalable multicast network system. Due to the wireless network and dynamic nature, secure communication in any multicast communication such as mobile ad hoc network (MANET) and vehicular ad hoc network (VANET) is highly important. The security function should have the capability to effectively manage any of the multicast networks. The main factors are credibility, integrity, and availability [10]. Authentication in t4erms of security is the capacity that depends on whether a peer unit in an association is the one that evidences its presence, or the data is used to determine its origin. Survival of network service depends on symmetric or asymmetric. The first is based on a shared secret key between two nodes that allows for safe communication, and the second is based on two separate sorts of keys, one private and the other public. The public key is used for encryption, and it is made public. Decryption is done mostly with the private key. Asymmetric cryptography necessitates arranging more resources than symmetric cryptography [11]. From any aspect, security is built on three pillars, as indicated in Figure 2.

The essential underpinnings of information security include parameters like confidentiality, integrity, and availability (CIA). Each security control and vulnerability can be analyzed using one or more of these basic ideas. Any security measure must appropriately address the entire CIA triad to be called comprehensive and complete.

2. Related Work

A large variety of multicast applications is available in wireless networks, but security is the main concern [12]. Moreover, the lack of security safeguards in multicast fields is a hindrance. The information can be shared by enabling access management cryptography in multicast applications. To encrypt group information, a shared key, also known as a traffic encryption key or a group key, is employed [13]. These keys are only accessible to authorized users; thus, they can only enter in groups. As a result, key management is an important part of secured wireless multicast. When ordinary text is encrypted, the key changes it to ciphertext, and vice versa when decrypted. Algorithms employ keys in a variety of ways. In practice, public cryptographic algorithms are widely used in traditional cryptocurrencies because of the difficulties in key distributions. The distribution of secret keys in a medium is required in such a manner that it does not affect any kind of information whether available, privately, or publicly, which is very important in both aspects as depicted in Figure 3.

A cryptographer attempts to create more and more sophisticated means of transmitting sensitive information, but hackers and code breakers work furiously to crack the system. System security is possible with the help of cryptography. This process of obtaining any information and the process of influencing the system using decryption [14] are an endless extended process.

2.1. Network and Keys Distributions. In wireless multicast communication, the three most significant aspects of key

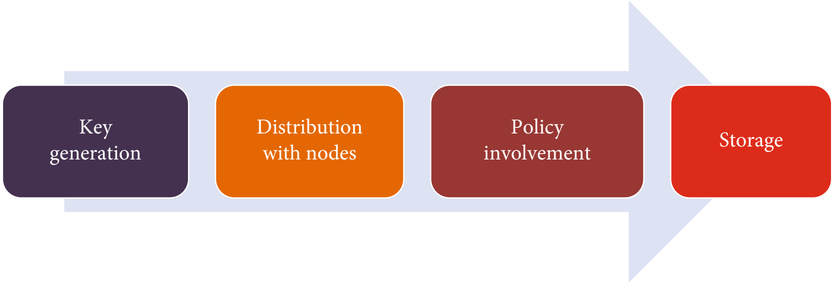


Figure 1: Distribution cycle.

Availability
Integrity
Confidentiality

Figure 2: Factors for network security.

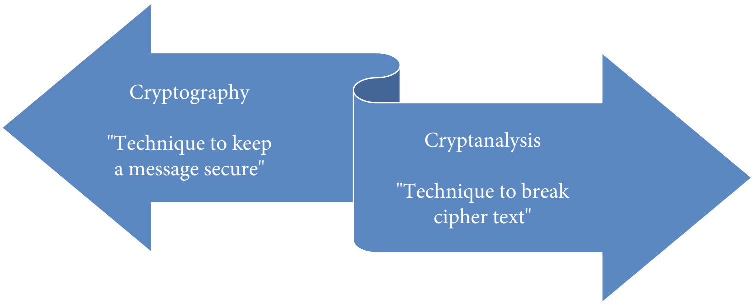

Cryptography
"Technique to keep a message secure"
Cryptanalysis
"Technique to break cipher text"

Figure 3: Cryptology segment.

management are key generation [15], key sharing, and key storage. Cryptography merely employs a mixed perspective to establish effective controls, particularly in hierarchical architecture. However, symmetric key and public key cause a huge exhibition center of cryptography. It is still attractive to give the impression of being a symmetric key-based improvement. Network topologies can be split into two types based on the motions for key management that are available: the hierarchical and nonhierarchical architecture of nodes [16]. A hierarchical network is usually the result of a symmetric key-based management protocol, which is a reasonable choice of nodes in the network. Experts recommend haphazard ways of key management in hierarchical networks, with no guarantees of successful key installation but the risk of a compromised node capacity. The network protocols require large storage space for key storage at each node [17]. Key distribution (KD) effectively supports a hierarchical network. KD makes it easier to generate and manage less flexible groups, since unlimited keys are used in the schemes which are based on the encryption methods and the basis of their key installation system. This plan is based on the Merkel Identity [18]. Through this scheme, one can implement information communication directly, and with it, any subgroup reduces the communication capacity, making the multicast communication scheme better and in the correct format [19].

2.2. Public Key Cryptography. In the 1970s, there were primarily two types of public key schemes discovered: the Diffie-Hellman agreement in 1975 and digital signature plans in 1977, Rivest, Shamir, and Edelman (RSA). A discrete logarithm problem underpins the D-H agreement scheme. The RSA encryption algorithm is based on a whole-number factorization problem, such as a number "n " is the result of two primes, "p" and "q" that are discovered. The hardness of the number factorization problem is crucial for security. After El-Gamal, public key encryption and signature schemes were matched by the integer factor in 1984 RSA, and then, elastic charging engine (ECE) came into existence [20].

2.2.1. Public Key Encryption. A public key encryption (PKI) polynomial is a kind of algorithm.

2.2.2. Key-Gen (1λ). A private key generator (PKG) takes a random key generation algorithm that uses a security parameter 1^λ as the input and the secret key (SK) and public key (PK) output.

2.2.3. Encryption (m, PK). Random encryption algorithm takes the message, PK input, and output ciphertext "C" from the public key.

2.2.4. Decryption (C, SK). The deterministic polynomial-time algorithm takes a ciphertext C and secret key SK of the receiver as input and outputs ciphertext "C."

2.3. Identity-Based Cryptography. An identity-based encryption (IBE) system can identify public key users with the public key system such as an email address [21].

2.3.1. Encryption Scheme Model. The first three-step can be random but not final, and it is deterministic. An identity-based encryption (IBE) scheme includes four steps of algorithms:

(1) Setup (1λ) is powered by a private key generator (PKG) that is a random polynomial-time algorithm for which input is 1λ

(2) Key-Gen (ID, MSK, and Params) is powered by PKG, a secret key Params output related to a random polynomial-time algorithm that identifies the master secret key secret and public parameter secret inputs

(3) Encryption m is a random polyalgorithm that takes the message m, the public key of sender $pkPKG$, and the public parameters Params of input and outputs ciphertext "C."

(4) Decryption (C, SKID, and Pa) is a deterministic polynomial-time algorithm that takes a ciphertext C [22]

The route efficiency to convey data to so many nodes in a network was determined using multicast communication based on an artificial neural network (ANN). Multicasting in MANETS handles concerns of security and quality of service (QoS), making this an excellent field for ANN implementation. The relationship between past, current, and future route discoveries of the distinct nodes in the mobility range can be discovered using ANN. The author proposed an innovative and practical use of ANN for secure multicast communication with supporting nodes. ANN consists of variable inputs used to determine the optimum number of neurons for the hidden layer by selecting the multicasting and supporting a node-routing function. The proposed model was based on the feedforward neural network (FFN) and backpropagation algorithms [23].

Fuzzy-based policies are also used to enhance the performance of the On-demand multicast routing protocol. The main objective is to establish a small, high-quality, and efficient forwarding group. Hence, the packet delivery rate also increases up to 40% and reduces the average end-to-end delay by about 35% [24]. There are several mechanisms of detection in distributed denial of service (DDoS) attacks over a multicast network. This attack affects the ongoing communication in the multicast network while also causing the wireless nodes to exhaust their energy much earlier than expected. This attack also results in a collision and minimal interference. A fuzzy-based system was designed to increase the reliability of attack detection [25]. Wireless sensor networks are also designed to provide various real-time applications. For providing energy-efficient transmissions, a congestion control mechanism is proposed at an optimized rate. The rate-based congestion control algorithm is based on cluster routing to offer minimum energy consumption. The rate control process reduces the end-to-end delay to improve network lifetime for a large simulation period [26].

To secure downlink multicast communication in edge-envisioned advanced metering infrastructure networks [27], a lightweight elliptic curve signcryption technique based on cipher text-policy feature-based encryption was proposed. The classic secret method maintains security by extending the length of keys [28], but it also raises the difficulty of calculation with the advancement of technology and cryptographic processing technology. As a result, creating a better encryption algorithm is a good way to ensure multicast communication. In the presence of multiple eavesdroppers, an intelligent reflecting surface (IRS) [29] is aided with the secure wireless powered communication network (WPCN), in which the transmitter uses the energy from a power station (PS), and that energy was used to multicast the transmit information to many IoT devices. Surface image security can be enhanced using artificial neural networks. Convolutional neural networks (CNNs) are useful to extract the features and information from hyperspectral images [30]. The deep spatial-spectral global reasoning network [31] takes into account both local and global information for hyperspectral images noise removal. Trust-based key management [32] is used to accomplish secure and efficient wireless multicast communication which can be applied for the security of destination-sequenced distance vector (DSDV), optimized link state routing (OLSR) [33], and ad hoc on-demand distance vector (AODV) [34] routing protocols. For device-to-device communication in the wireless system, the delay, memory, and hardware resources utilization [35, 36] are a major concern. It has been identified in different topological communication [37] such as in Zigbee IEEE 802.14 [38, 39], wireless sensor network, network-on-chip communication, wireless monitoring of plant information, and security. Users require wireless connectivity regardless of their geographic location; hence, mobile ad hoc networks are gaining popularity at an all-time high. Mobile ad hoc networks [40] are becoming more vulnerable to security threats. MANETs must use a secure manner of communication and transmission, which is a difficult and time-consuming task. Researchers worked specifically on the security challenges in MANETs to enable safe transmission and communication. Fuzzy adaptive data transmission congestion prediction [41] is used to increase network stability since traffic congestion is widespread in multimedia networks. A fuzzy adaptive prediction solution for data transmission congestion has been developed in multimedia networks. The unique approach of fuzzification-defuzzification has been proposed in the paper to support multicast communication and cryptography with different parameters in the wireless communication system.

3. Proposed Work

In the paper, we propose the implementation of key distribution based on a fuzzy set of rules to generate random keys.

These methods are based on logical AND, logical OR, and logical AND-OR rules. ANN is used in cryptography, used to generate strong cipher, and offers less overhead. The main aim of the research work is to build an encryption system based on fuzzy logic to secure confidentiality, availability, and integrity in the key management of wireless multicast communication. The principles of symmetric cryptography with fuzzy-based rules are applied to encrypt information. As we studied in the previous work, it was observed that the fuzzy IBE scheme is sensitive and offers security only for selective-ID attacks in very few models. However, this scheme is secure as long as one hashes the identity before using it. Currently, there is no fuzzy IBE available that is indistinguishable under an adaptive ciphertext attack (CCA2) secure. Therefore, a new fuzzy IBE scheme is suggested to achieve CCA2 security based on public key parameters whose size is not dependent on the number of attributes associated with an identity.

The research work is focused on secured wireless communication using fuzzy logic-based high-speed symmetric key cryptographic key management methods that have been proposed to addresses the main issues like computational safety, power reduction, and less memory in multicast communication and also covers CIA.

Though conventional methods of cryptography work on the digital values, i.e., 0 or 1, here proposed methods are based on fuzzy values of key distribution parameters like initial, mid, low, and high, which offers more accurate constraints for security pillars. Though conventional cryptography methods are a sort of public key cryptography used in wireless multicast communication that provides an equivalent level of security with higher overheads, fuzzy-based offer reduced computation and storage overheads. In comparison to the previous fuzzy IBE schemes, our scheme has short parameters and a tight reduction simultaneously. This method offers a shorter computational time for keys, reduced power consumption, and limited usage of memory without compromising the CIA attributes.

3.1. Fuzzy Implementation. Fuzzy logic is based on critical thinking. To accomplish security, the algorithm utilizes variables. It features a diverse key structure of up to 128-bit. The client can define the key in the correct format in it that is fixed as a secret key. It includes a method that is similar to human reasoning, and possible digital values are “0,” “1,” and intermediate [18]. If we do so and do not believe the two Boolean values, fuzzy logic may accept any of them as “yes,” “it is conceivable,” “of course,” “we cannot say,” “not possible,” and “definitely not.” It helps in dealing with the uncertainty of various areas.

3.1.1. Parameters for Key Distribution in Symmetric Key Perspective. In the symmetric random set approach based on fuzzy logic, all valid keys make a key pool and each party will set up a set of keys pool randomly. The type of the key pool housed in each member is selected properly. A key group of members is very likely, i.e., sharing it. In our approach, we are taking advantage of this feature, i.e., when authentication went to share a pool, keys are provided for symmetric key distribution for the purpose, and RSA separates symmetric key distribution by sending a key request, as other parties may also provide the key. Following notation and assumptions are used for key distribution using symmetric key protocols.

(1) Parties/Principles (A, B, S, and E). We assume the two parties who wish to agree on a secret are A and B, while a trusted third party is S and an attacker is E.

(2) Shared Secret Keys. Kab, Kbs, Kas, and Kab denote a secret key known only to A and B.

(3) Nonces (M, N, N_a, and N_b). Nonces are random numbers. Na denotes a nonce originally produced by the principle A.

(4) Timestamps (Ta, Tb, and Ts). Ta is the time stamp produced by A. Timestamp is used for synchronization.

3.1.2. Logical AND-OR-Based Rule. The fuzzy rules set for various parameters for key distribution are decided based on an AND-OR logic. The parameters may take any value in the range of initial, min, mid, and high. A key distribution policy is observed by setting various combinations of parameter values.

3.1.3. AND Rules-Based Algorithm

(i) If (Parties/Principal is normal) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

(ii) If (Parties/Principal is High) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_High) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is High_key) (1)

(iii) If (Parties/Principal is Initial) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_High) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is High_key) (1)

(iv) If (Parties/Principal is Initial) and (Shared_Secret_Keys is Min_Shared) and (Nonces is Nonces_High) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is High_key) (1)

(v) If (Parties/Principal is Initial) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_High) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is High_key) (1)

(vi) If (Parties/Principal is Initial) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_High) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is Mid_key) (1)

(vii) If (Parties/Principal is High) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_

Step 1 (variables declarations): in this step, we will select the most prominent variable that affects the key distribution (parties/principals, shared secret keys, etc.)
Step 2 (fuzzification): this is the most important step of our method. This step is itself divided into two parts, i.e., fuzzification and defuzzification. This will help us to convert the fractional values into "0" and "1" values
Step 3 (rule implementations): rule preparation is based on the logical AND of each variable involve and its impact on the final predicted value. Same as that logical OR, each variable and its impact on the final predicted value are involved
Step 4 (convert to graph): this graph helps show the rise and fall of the final output on the 3D surface

Algorithm 1: Fuzzy implementation algorithm.

Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

(viii) If (Parties/Principal is High) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

(ix) If (Parties/Principal is High) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_High) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is High_key) (1)

(x) If (Parties/Principal is High) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Final_Time_stanp), then (Key_distribution_policy is Mid_key) (1)

(xi) If (Parties/Principal is normal) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

The algorithm provides a set of AND rules that are prepared to show the logical summation of all the possible effects of the key distribution policy, as listed in Table 1.

Similarly, the set of OR rules is prepared that show the logical OR-based summation of all the possible effects on key distribution policy, which is given below.

3.1.4. OR Rules-Based Algorithm

(i) If (Parties/Principal is Initial) or (Shared_Secret_Keys is Min_Shared) or (Nonces is Nonces_Min) or (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is High_key) (1)

(ii) If (Parties/Principal is Initial) or (Shared_Secret_Keys is Mid_Shared) or (Nonces is Nonces_Min) or (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Mid_key) (1)

(iii) If (Parties/Principal is Initial) or (Shared_Secret_Keys is Mid_Shared) or (Nonces is Nonces_Mid) or (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Less_Key) (1)

(iv) If (Parties/Principal is Initial) or (Shared_Secret_Keys is Mid_Shared) or (Nonces is Nonces_Mid) or (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

(v) If (Parties/Principal is Initial) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_High) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is High_key) (1)

(vi) If (Parties/Principal is normal) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is High_key) (1)

(vii) If (Parties/Principal is High) and (Shared_Secret_Keys is Min_Shared) and (Nonces is Nonces_High) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Mid_key) (1)

(viii) If (Parties/Principal is High) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Less_Key) (1)

(ix) If (Parties/Principal is Initial) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Less_Key) (1)

(x) If (Parties/Principal is Initial) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Min) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Less_Key) (1)

(xi) If (Parties/Principal is Initial) and (Shared_Secret_Keys is High_Shared) and (Nonces is Nonces_Min) and (Timestamps is Inital_Time_stanp), then (Key_distribution_policy is Less_Key) (1)

(xii) If (Parties/Principal is normal) and (Shared_Secret_Keys is Mid_Shared) and (Nonces is Nonces_Mid) and (Timestamps is Mid_TIme_stamp), then (Key_distribution_policy is Mid_key) (1)

Fuzzy logic is a measure of the membership of certainty or uncertainty of the elements of a set that were chosen. Key distribution rules are decided based on the principle, which are defined by the fuzzy logic for similar cases, as listed in Table 2.

Table 1: Key distribution based on logical AND.

1.	(Parties/Principal is initial)	(Shared_Secret_Keys is Min_Shared)	(Nonces is Nonces_Min)	(Timestamps is Inital_Time_stamp)	(Key_distribution_policy is Less_Key)
2.	(Parties/Principal is normal)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policy is Mid_key)
3.	(Parties/Principal is high)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_High)	(Timestamps is Final_Time_stanp)	Key_distribution_policyisHigh_key)
4.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_High)	(Timestamps is Final_Time_stanp)	(Key_distribution_policyisHigh_key)
5.	(Parties/Principal is initial)	(Shared_Secret_Keys is Min_Shared)	(Nonces is Nonces_High)	(Timestamps is Final_Time_stanp)	(Key_distribution_policyisHigh_key)
6.	(Parties/Principal is initial)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_High)	(Timestamps is Final_Time_stanp)	(Key_distribution_policyisHigh_key)
7.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_High)	(Timestamps is Final_Time_stanp)	(Key_distribution_policy is Mid_key)
8.	(Parties/Principal is high)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policy is Mid_key)
9.	(Parties/Principal is high)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policyisMid_key)
10.	(Parties/Principal is high)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_High)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policyisHigh_key)
11.	(Parties/Principal is high)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Final_Time_stamp)	(Key_distribution_policy is Mid_key)
12.	(Parties/Principal is normal)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_Time_stamp)	(Key_distribution_policy is Mid_key)

Table 2: Key distribution based on logical OR.

1.	(Parties/Principal is initial)	(Shared_Secret_Keys is Min_Shared)	(Nonces is Nonces_Min)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is High_key)
2.	(Parties/Principal is initial)	Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Min)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Mid_key)
3.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Less_Key)
4.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policy is Mid_key)
5.	(Parties/Principal is initial)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_High)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is High_key)
6.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policy is High_key)
7.	(Parties/Principal is high)	(Shared_Secret_Keys is Min_Shared)	(Nonces is Nonces_High)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Mid_key)
8.	(Parties/Principal is high)	(Shared_Secret_Keys is Mid_Shared	(Nonces is Nonces_Mid)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Less_Key)
9.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Less_Key)
10.	(Parties/Principal is initial)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Min)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Less_Key)
11.	(Parties/Principal is initial)	(Shared_Secret_Keys is High_Shared)	(Nonces is Nonces_Min)	(Timestamps is Inital_Time_stanp)	(Key_distribution_policy is Less_Key)
12.	(Parties/Principal is normal)	(Shared_Secret_Keys is Mid_Shared)	(Nonces is Nonces_Mid)	(Timestamps is Mid_TIme_stamp)	(Key_distribution_policy is Mid_key)

It is found that OR- and AND-fuzzy algorithms are strong, in the sense that they are not very sensitive to the changing environment and misplaced or forgotten the rules. Because computational logic is generally considerably simpler than exact systems logic, it uses less processing power.

4. Results and Discussions

MATLAB 2018 is used to model and simulate the fuzzy logic control system. It consists of the fuzzy logic toolbox that gives a fuzzy controller block in Simulink. This toolbox provides a fuzzy interface (FIS) editor, membership function editor, rule editor, rule viewer, and surface viewer. Simulink is an environment based on block representations that help in modeling, simulation, and analysis.

4.1. Implementation of Fuzzification Rules-Based Models and Results. Fuzzy inference is the process of formulating the mapping from given input for key distribution to an output using fuzzy logic-based rules.

The mapping (key distribution) provides a basic platform for decision-making and patterns. The process of fuzzy inference involves all the pieces that are described in membership functions, logical operations, and if-then rules [26]. After implementing the encryption algorithm, the results are presented at various levels.

Figure 4 shows the AND rule-based member function. Four inputs are inserted as parties with the shared keys command for the key distribution function and the output is in the 3D surface graph. Logical key distribution applies the implication and aggregation of variables. Figure 5 highlights the second structure for key distribution in which three-axis are used. The x-axis, y-axis, and z-axis are for parities, shared key loss integrity, and output of key distribution policy, respectively. Figure 6 shows the 3D structure for key distribution based on OR rules. The x-axis, y-axis, and z-axis are for shared secret keys, parties and principle loss integrity, and constant output using key distribution policy, respectively.

4.2. Parameters Used for Fuzzy-Based Model for Key Compromise Prediction. The fuzzy model for the key compromise prediction technique is based on the following policy [26].

4.2.1. Loss of Confidentiality. The control and encryption of members are in a fixed format that works to prevent damage to its structure. As an example, users will first need to install, and then users can access based on their proven identity. Allowed users may only provide access to data in a detailed format. If users are not allowed, they are denied access.

4.2.2. Loss of Integrity. The general way of determining the nearness of hashing is fixed. In a detailed format, the estimation can be done with the help of a hashing algorithm for a certain file or data string.

4.2.3. Loss of Authentication. Authentication is an important security aspect. The public key and symmetric key cryptography can provide those services. Symmetric key cryptography with message authentication (MAC) only provides evidence that one of the parties is associated with the shared key.

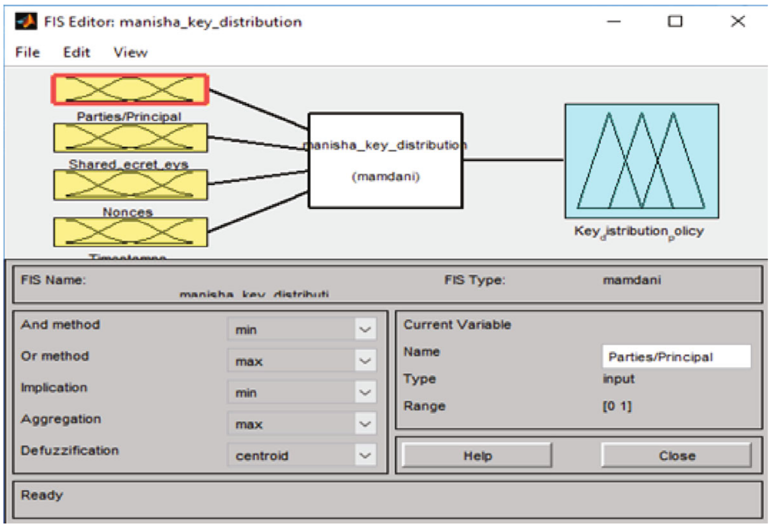
4.2.4. Loss of Nonrepudiation. It is assurance that the party cannot deny the validity of anything.

4.3. Surface Generation. Figure 7 shows the 3D structure of surface_1 for key distribution based on the AND-OR model. The x-axis represents the loss of confidentiality, the y-axis represents the loss of integrity, and the z-axis represents the constant key compromised. In this segment, we found two-fragment evaluations such as integrity and confidentiality of key loss distribution. It gives key distribution based on the AND-OR model for integrity and confidentiality analysis.

In the same way, Figure 8 shows the suface_2 plot of loss of security key graphical charts from left to right. The distribution key loss is counted with security parameters from the right to the left from the end of distribution to its beginning. Typically, the distribution ends with two events such as confidentiality and authentication. Figure 9 depicts the surface_ 3 plot that shows the loss security key graphical charts from left to right. The distribution key loss is counted with security parameters from the right to the left and the end of distribution to its beginning. Typically, the distribution ends with two events such as confidentiality and nonrepudiation. Figure 10 shows the surface_4 structure for key distribution, in which the x-axis is used for loss of authentication, the y -axis for loss of nonrepudiation, and the z-axis as constant for key compromised. In this segment, two fragments are considered for evaluation such as authentication and nonrepudiation of the key loss distribution.

The data communication is verified among the different nodes in AODV routing to support the multicast routing in the wireless networking for effective cryptographic communication with the secured key. The simulation is done to verify the encryption and decryption of the plain text with the variable key length of 8-bit to 128-bit. Test case-1 and test case-2 summarize the test inputs for input plain text, ciphertext, and decrypted text. The communication system supports 100 nodes (M_0 to M_{99}). The source nodes have the plain text, encryption key as the inputs, and ciphertext as the output. In the encryption end, the destination nodes have the ciphertext, decryption key as inputs, and decrypted text as output.

4.3.1. Test Case 1 (64-Bit). The data communication is verified from the source node M_9 to destination node M_{99}: Text_in (64 bits) = "01001001 01101110 01100100 01101001 01100001 00110001 00110010 00110011"(binary) = 49 6E 64 69 61 31 32 33 (hexadecimal) = India123 (in ASCII), Enecryption_Key_Gen (64 bits) = "01001101 01100001 01101110 01101001 01110011 01101000 01100001 01000000" (binary) = 4D 61 6E 69 73 68 61 40 (hexadecimal) = Manisha@(in ASCII), Cipher_text (64 bits) = "00000100 00001111 00001010 00000000 00001000 01011001 01010011 01110011" (binary) = 04 0F 0A 00 08 59 53 73 (hexadecimal) = □□□□□YSs (in ASCII), Decryption_Key_Gen (64 bits) = "01001101 01100001 01101110 01101001 01110011 01101000 01100001 01000000" = 1'h (4D 61 6E 69 73 68 61 40) = Manisha@(in ASCII), and Decrypted_text = " 01001001 01101110 01100100 01101001 01100001 00110001 00110010 00110011"(binary) = 49 6E 64 69 61 31 32 33 (hexadecimal) = India123 (in ASCII).



FIS Name: manisha_key_distributi

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

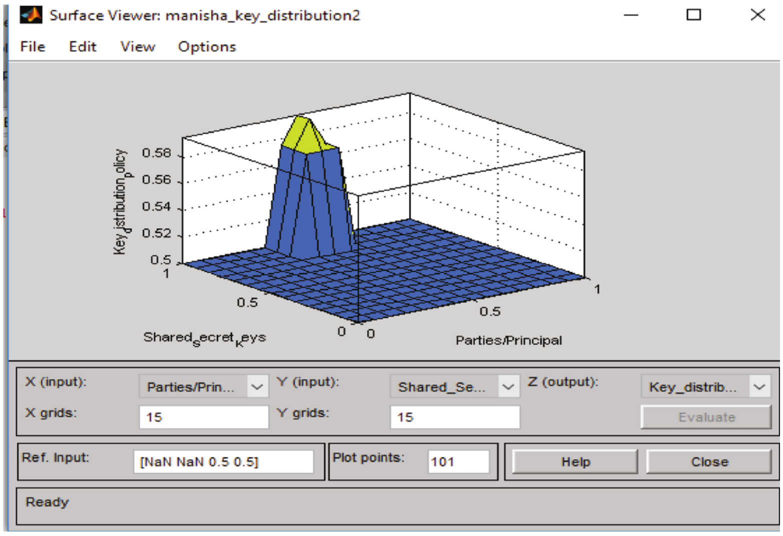
Name: Parties/Principal

Type: input

Range: [0 1]

Ready

Figure 4: AND rule member function.



X (input): Parties/Prin...

Y (input): Shared_Se...

Z (output): Key_distrib...

X grids: 15

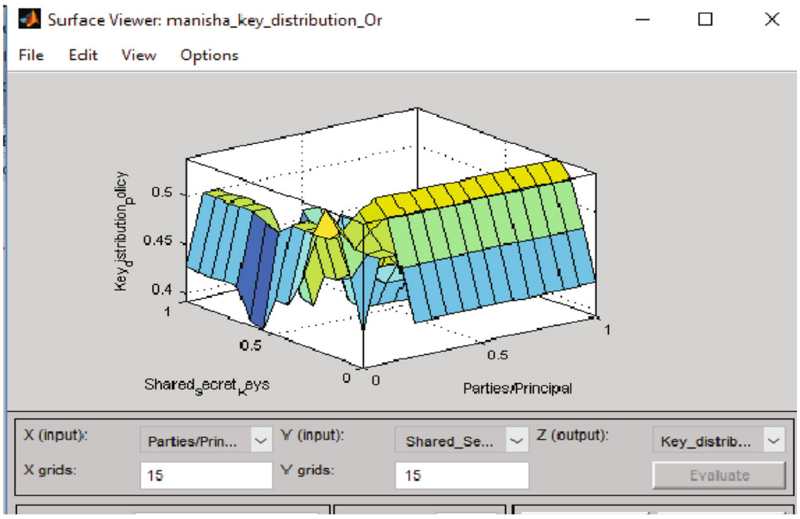
Y grids: 15

Ref. Input: [NaN NaN 0.5 0.5]

Plot points: 101

Ready

Figure 5: Surface evaluation of logical AND membership.



X (input): Parties/Prin...

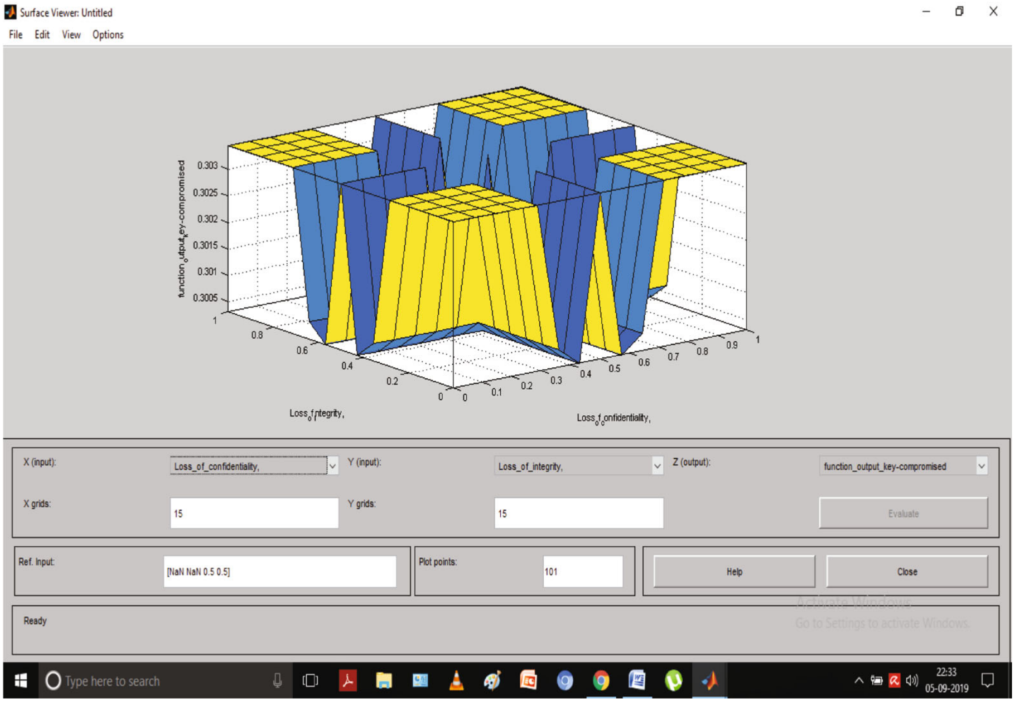
Y (input): Shared_Se...

Z (output): Key_distrib...

X grids: 15

Y grids: 15

Figure 6: Surface evaluation of logical OR rule membership.



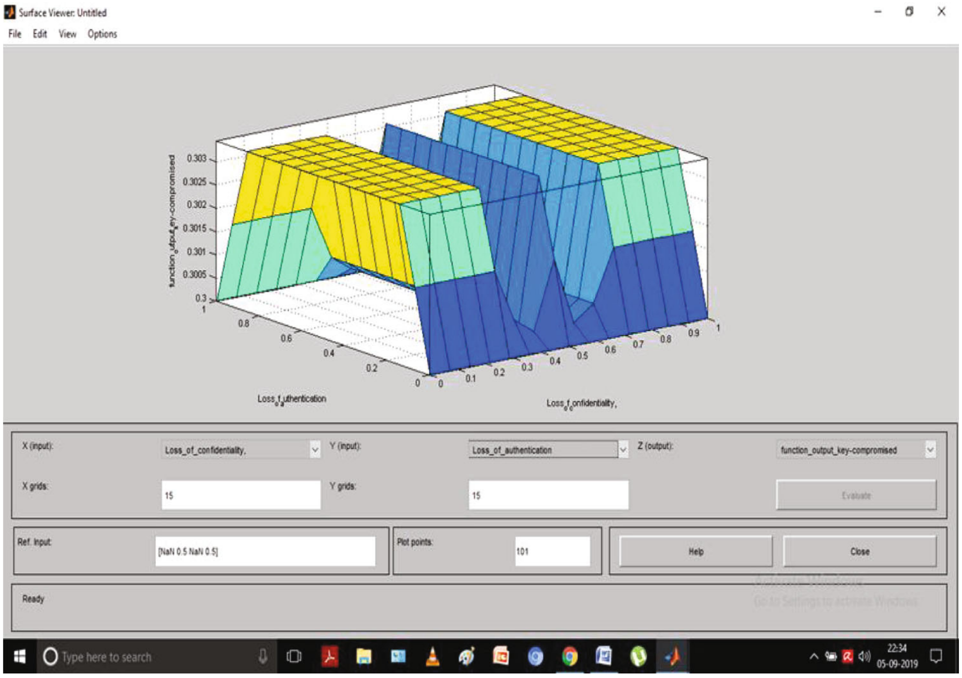
X (input): Loss_of_confidentiality, Y (input): Loss_of_integrity, Z (output): function_output_key-compromised

X grids: 15 Y grids: 15

Ref. Input: [NaN NaN 0.5 0.5] Plot points: 101

Ready

Figure 7: Surface_1 for integrity and confidentiality analysis.



X (input): Loss_of_confidentiality, Y (input): Loss_of_authentication Z (output): function_output_key-compromised

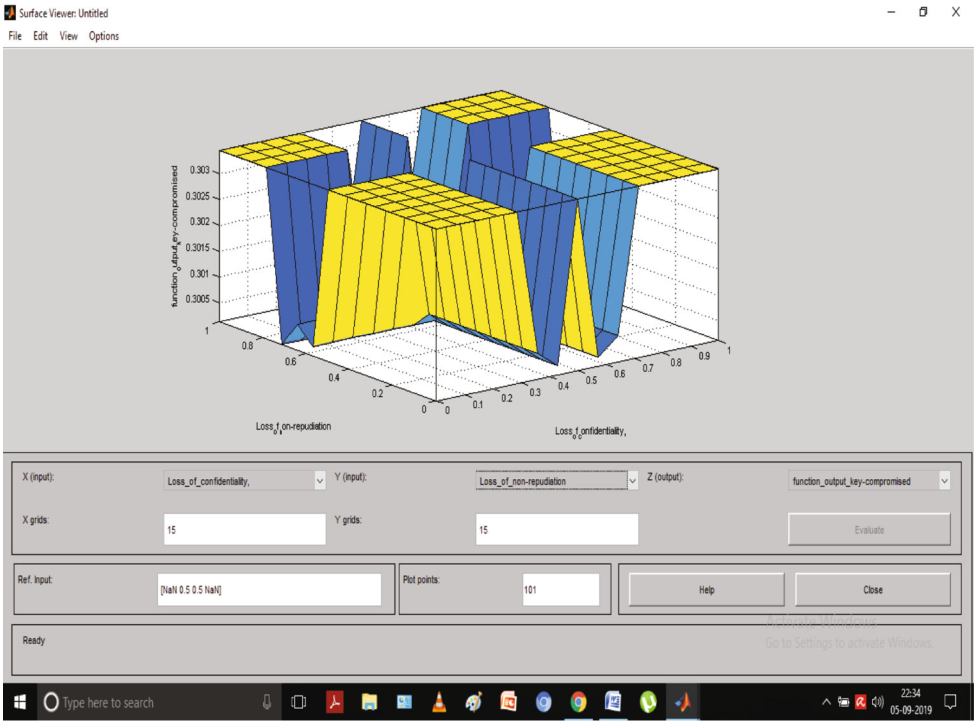
X grids: 15 Y grids: 15

Ref. Input: [NaN 0.5 NaN 0.5] Plot points: 101

Ready

Figure 8: Surface_2 for confidentiality and authentication analysis.

4.3.2. Test Case 2 (128-Bit). The data communication is verified from the source node M_6 to destination node M_{90}: Text_in (128 bits) = "01000101 01101100 01100101 01100011 01110100 01110010 01101111 01101110 01101001 01100011 01110011 01000000 00110001 00110010 00110011 00110100" (binary) = 45 6C 65 63 74 72 6F 6E 69 63 73 40 31 32 33 34 (hexadecimal) = Electronics@1234 (in ASCII), Enecryption_Key_Gen (128 bits) = "01001101 01100001 01101110 01101001 01110011 01101000 01100001 01111001 01100001 01100100 01100001 01110110 01000000 00110001 00110010 00110011" = 4D 61 6E 69 73 68 61 79 61 64 61 76 40 31 32 33 (hexadecimal) = Manishayadav@123 (in ASCII), Cipher_text (128 bits) = "00001000 00001101 00001011 00001010 00000111 00011010 00001110 00010111 00001000



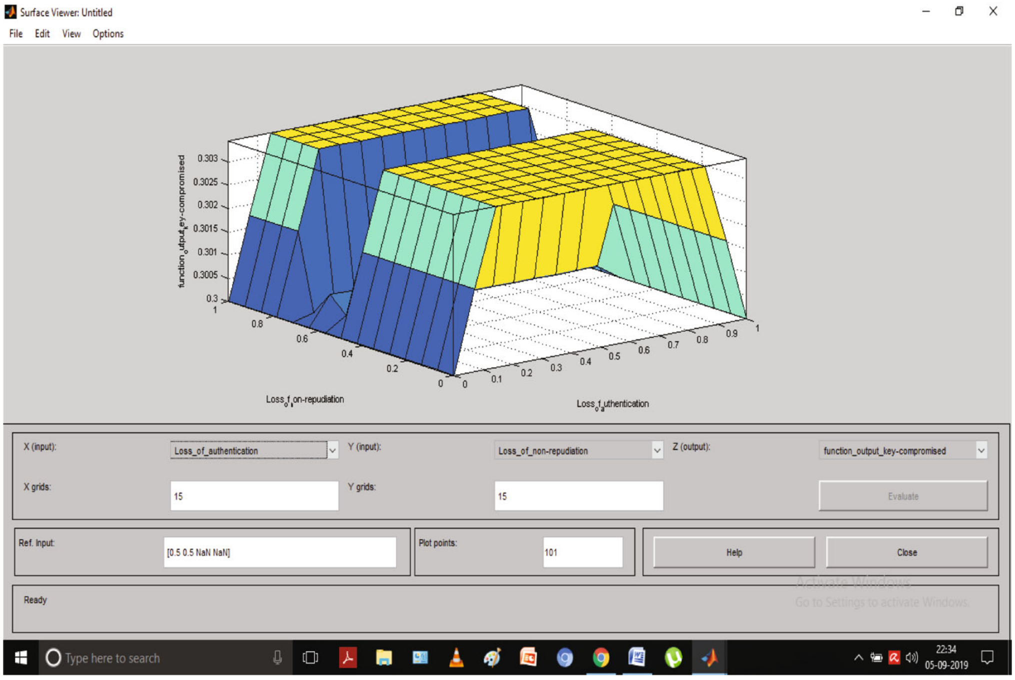
X (input): Loss_of_confidentiality, Y (input): Loss_of_non-repudiation Z (output): function_output_key-compromised

X grids: 15 Y grids: 15

Ref. Input: [NaN 0.5 0.5 NaN] Plot points: 101

Ready

Figure 9: Surface_3 for confidentiality and nonrepudiation analysis.



X (input): Loss_of_authentication Y (input): Loss_of_non-repudiation Z (output): function_output_key-compromised

X grids: 15 Y grids: 15

Ref. Input: [0.5 0.5 NaN NaN] Plot points: 101

Ready

Figure 10: Surface_4 for authentication and nonrepudiation analysis.

00000111 00010010 00110110 01110001 00000011 00000001 00000111” = 08 0D 0B 0A 07 1A 0E 17 08 07 12 36 71 03 01 07 (hexadecimal) = □□□□□□□□□□□6q□□□(in ASCII), Decryption_Key_Gen (128 bits) = “01001101 01100001 01101110 01101001 01110011 01101000 01100001 01111001 01100001 01100100 01100001 01110110 01000000 00110001 00110010 00110011” = 4D 61 6E 69 73 68 61 79 61 64 61 76 40 31 32 33 = Manishayadav@123 (in ASCII), and Decrypted_text = “01000101 01101100 01100101 01100011 01110100 01110010 01101111 01101110 01101001 01100011 01110011

Table 3: Performance parameters for AODV.

	Delay (sec)	Throughput (bps)	Control overhead	PDR
10	0.037	0.210	0.0057	0.980
20	0.085	0.195	0.0058	0.978
30	0.091	0.181	0.0061	0.996
40	0.142	0.172	0.0058	0.985
50	0.196	0.158	0.0048	0.983
60	0.289	0.152	0.0058	0.995
70	0.324	0.172	0.0053	0.991
80	0.349	0.191	0.0040	0.986
90	0.397	0.197	0.0035	0.975
100	0.410	0.225	0.0065	0.989

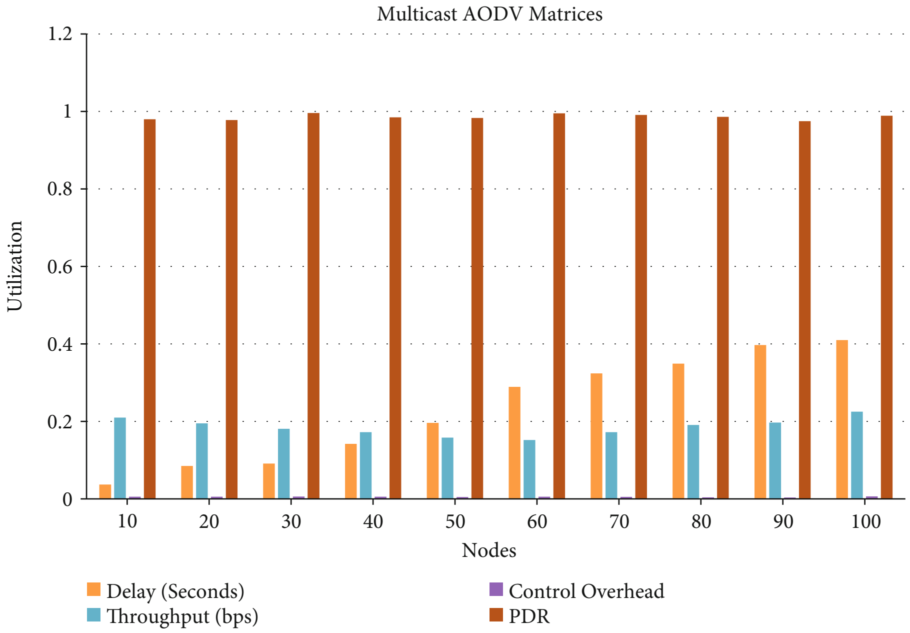


Figure 11: Multicast nodes variations and parameters.

01000000 00110001 00110010 00110011 00110100” (binary) = 45 6C 65 63 74 72 6F 6E 69 63 73 40 31 32 33 34 (hexadecimal) = Electronics@1234 (in ASCII).

The performance of the multicast system is evaluated based on the different performance indices such as end-to-end delay, throughput, packet delivery ratio, and control overhead. Table 3 lists the values of the parameters. Figure 11 presents the graph corresponding to the analysis. It has been analyzed that the delay is increasing with the nodes. The packet delivery ratio is good with optimal values of control overhead and throughput.

5. Conclusions

The research work provides a novel way of key distribution using a fuzzy-based cryptography model. To anticipate the key distribution in symmetric key cryptography, initially, the AND logic is created, and rule sets are prepared. Then, using the same parameters, the OR logic is created. We used a general data set of 20-25 values for each set to set the formulation of the generic rules. A 3D surface graph has been created based on these models, and these graphs are useful in demonstrating the challenges of associated factors and their final effect on key distribution. Various security features are analyzed because of the implementation of these factors. The cryptographic encryption and decryption of the data are verified successfully based on AODV routing in wireless communication. The large key size provides greater security. The 64-bit and 128-bit key encryption and decryption are experienced with 64-bit and 128-bit plain text. The multicast system supports 100 nodes of data interchange with minimum overhead, delay, and maximum throughput. In the future, we are planning to integrate the system with field-programmable gate array (FPGA) hardware with a larger key size and data. The number of nodes can be enhanced to evaluate the performance of the system for large scalable networks and efficient multicast communication.

Data Availability

There is no external data used in the paper. The data is available upon request to the corresponding author.

Disclosure

Manisha Yadav is working at Dr. R.M.L University, Ayodhya, and doing a Ph.D. research work at Dr. APJ Abdul Kalam Technical University, Lucknow, India.

Conflicts of Interest

The authors declare that they have no conflicts of interest to report regarding the present study.

Acknowledgments

The work is supported by the Secure Computing Lab, School of Computer and System Sciences, Jawaharlal Nehru University, New Delhi, under DST project 115 DDP/TDP/12.

References

[1] A. D. Borkar and M. Atulkar, "Fuzzy inference system for image processing," *International Journal of Advanced Research in Computer Engineering and Technology*, vol. 2, no. 3, pp. 1007–1010, 2013.

[2] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, "Measurements of a wireless link in an industrial environment using an IEEE 802.11compliant physical layer," *IEEE Transactions on Industrial Electronics and Control Instrumentation*, vol. 49, no. 6, pp. 1265–1282, 2002.

[3] A. R. Sattam and P. Kenneth, "Certificate less public key cryptography a full version," *Lecture Notes in Computer Science*, vol. 2894, pp. 452–473, 2003.

[4] P. S. L. M. Barreto, H. Y. Kim, B. Lynn, and M. Scott, "Efficient algorithms for pairing-based cryptosystems," *Lecture Notes in Computer Science*, vol. 2442, pp. 354–368, 2002.

[5] D. Das, U. A. Lanjewar, and S. J. Sharma, "The art of cryptology: from ancient number system to strange number system," *International Journal of Industrial Engineering and Management*, vol. 2, no. 4, pp. 265–275, 2013.

[6] P. Vishnoi, S. Shimi, and A. Kumar, "Symmetric cryptography and hardware chip implementation on FPGA," *Advances in Intelligent Systems and Computing*, vol. 989, pp. 945–955, 2020.

[7] S. Tayal and D. N. Gupta, "A review paper on network security and cryptography," *Advances in Computational Sciences and Technology*, vol. 10, no. 5, pp. 763–770, 2017.

[8] A. Kumar, P. Vishnoi, and S. L. Shimmi, "Smart grid security with cryptographic chip integration," *EAI Endorsed Transactions on Energy Web*, vol. 6, no. 23, pp. 1–12, 2019.

[9] M. Manjul, R. Mishra, K. Singh, L. H. Son, M. A. Basset, and P. H. Thong, "Single rate based extended logarithmic multicast congestion control," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2779–2991, 2019.

[10] M. Agrawal and P. Mishra, "A comparative survey on symmetric key encryption techniques," *International Journal on Computer Science and Engineering*, vol. 4, no. 5, pp. 877–882, 2018.

[11] N. Balaji, P. Gurunathan, and A. Shanmugam, "Performance comparison of multicast routing protocols under variable bit rate scenario for mobile ad-hoc networks," *Journal of Network Security and Applications Communications in Computer and Information Science*, vol. 89, pp. 114–122, 2010.

[12] E. Thmbiraja, G. Ramesh, and R. Umarani, "A survey on various most common encryption techniques," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 2, no. 7, pp. 226–233, 2012.

[13] F. Liu and C. Wu, "Embedded extended visual cryptography schemes," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 307–322, 2011.

[14] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[15] X. Zhang, H. S. Seo, and C. Wang, "A lightweight encryption method for privacy protection in surveillance videos," *IEEE Access*, vol. 8, pp. 18074–18087, 2018.

[16] I. Dohare and K. Singh, "Green communication in sensor-enabled IoT: integrated physics-inspired meta-heuristic optimization-based approach," *Journal Wireless Networks*, vol. 26, no. 5, pp. 3331–3348, 2020.

[17] J. S. Kiran, M. Anusha, A. Vijaykumar, and M. Kavya, "Cryptography: the science of secure communication," *International Journal of Computer Science and Network Security*, vol. 16, no. 4, pp. 129–135, 2016.

[18] H. M. Mudia and P. V. Chavan, "Fuzzy logic based image encryption for confidential data transfer using (2, 2) secret sharing scheme," *Procedia Computer Science*, vol. 78, pp. 632–639, 2016.

[19] J. Jiang, Y. Liu, Z. Su, G. Zhang, and S. Xing, "An improved selective encryption for H.264 video based on intra prediction mode scrambling," *Journal of Multimedia*, vol. 5, no. 5, pp. 464–472, 2010.

[20] R. Sharma, "Study of latest emerging trends on cyber security and its challenges to society," *International Journal of Scientific & Engineering Research*, vol. 3, no. 6, pp. 1–4, 2012.

[21] R. Pahal and V. Kumar, "Efficient implementation of AES," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 7, pp. 290–295, 2013.

[22] S. Sattam and Al-Riyam, "Certificate less public key cryptography," in *Proceedings of the ASIACRYPT*, pp. 452–473, Taipei, Taiwan, 2003.

[23] B. S. Vishwanath, H. E. Naidu, K. Thanushkodi, M. B. S. Pandey, and G. Vasanth, "The novel application of artificial neural networks for a reliable secure wireless multicast routing in mobile ad-hock networks," in *Proceedings of the ICNVS*, pp. 54–62, India, 2017.

[24] H. Zhang, Z. Zhang, and H. Dai, "Gossip-based information spreading in mobile networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 11, pp. 5918–5928, 2013.

[25] G. Santhi and A. Nachiappan, "Fuzzy cost based multicast routing for mobile ad-hoc networks with improved QoS," *Advances in Digital Image Processing and Information Technology*, vol. 205, pp. 429–437, 2011.

[26] H. Cheng, Sheng, and X. Yang, "Hyper- mutation based genetic algorithms for dynamic multicast routing problem in mobile ADHOC networks," in *Proceedings of the TRUSTCOM*, pp. 1586–1592, Liverpool, UK, 2012.

[27] S. Khasawneh and M. Kadoch, "ECS-CP-ABE: a lightweight elliptic curve signcryption scheme based on ciphertext-policy

attribute-based encryption to secure downlink multicast communication in edge envisioned advanced metering infrastructure networks," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 8, p. 4102, 2021.

[28] Y. Liu and S. Zhang, "Research and implementation of secure multicast communication algorithm based on chaos neural network," *Innovative Computing*, pp. 1767–1771, 2022.

[29] L. Zhai, Y. Zou, J. Zhu, and B. Li, "Improving physical layer security in IRS-aided WPCN multicast systems via Stackelberg game," *IEEE Transactions on Communications*, 2022.

[30] D. Hong, Z. Han, J. Yao et al., "SpectralFormer: rethinking hyperspectral image classification with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[31] X. Cao, X. Fu, C. Xu, and D. Meng, "Deep spatial-spectral global reasoning network for hyperspectral image denoising," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022.

[32] M. Yadava, A. S. Pandey, and K. Singh, "Secure and efficient wireless multicast communication using trust-based key management," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 24, no. 3, pp. 711–727, 2021.

[33] N. Gupta, A. Jain, K. S. Vaisla, A. Kumar, and R. Kumar, "Performance analysis of DSDV and OLSR wireless sensor network routing protocols using FPGA hardware and machine learning," *Multimedia Tools and Applications*, vol. 80, no. 14, pp. 22301–22319, 2021.

[34] N. Gupta, K. S. Vaisla, A. Jain, A. Kumar, and R. Kumar, "Performance analysis of AODV routing for wireless sensor network in FPGA hardware," *Computer Systems Science and Engineering*, vol. 39, no. 2, pp. 1–12, 2021.

[35] N. Kumar, V. M. Mishra, and A. Kumar, "Smart grid and nuclear power plant security by integrating cryptographic hardware chip," *Nuclear Engineering and Technology*, vol. 53, no. 10, pp. 3327–3334, 2021.

[36] N. Kumar, V. M. Mishra, and A. Kumar, "Smart grid security with AES hardware chip," *International Journal of Information Technology*, vol. 12, no. 1, pp. 49–55, 2020.

[37] V. Ompal, M. Mishra, and A. Kumar, "FPGA integrated IEEE 802.15.4 ZigBee wireless sensor nodes performance for industrial plant monitoring and automation," *Nuclear Engineering and Technology*, 2022.

[38] V. Ompal, M. Mishra, and A. Kumar, "Zigbee internode communication and FPGA synthesis using mesh, star and cluster tree topological chip," *Wireless Personal Communications*, vol. 119, no. 2, pp. 1321–1339, 2021.

[39] A. Kumar, P. Kuchhal, and S. Singhal, "Secured network on chip (NoC) architecture and routing with modified TACIT cryptographic technique," *Procedia Computer Science*, vol. 48, pp. 158–165, 2015.

[40] P. Singh, M. Khari, and S. Vimal, "EESSMT: an energy efficient hybrid scheme for securing mobile ad hoc networks using IoT," *Wireless Personal Communications*, pp. 1–25, 2021.

[41] H. Liu and Y. C. Ko, "Fuzzy self-adaptive prediction method for data transmission congestion of multimedia network," *Wireless Networks*, pp. 1–10, 2021.