Ruzicka Indexed Regressive Homomorphic Ephemeral Key Benaloh Cryptography for Secure Data Aggregation in WSN

Saravanakumar Pichumani[1], T. V. P. Sundararajan[2], Rajesh Kumar Dhanaraj[3], Yunyoung Nam[4], Seifedine Kadry[5]

[1] Department of Electronics and Communication Engineering, Bannari Amman Institute of Technology, India
[2] Department of Electronics and Communication Engineering, Shri Shakthi Institute of Engineering and Technology, India
[3] Galgotias University, India
[4] Department of Computer Science and Engineering, Soonchunhyang University, South Korea
[5] Department of Applied Data Science, Noroff University College, Kristiansand, Norway
saravanakumarp@bitsathy.ac.in, suntvp@yahoo.co.in, sangeraje@gmail.com, ynam@sch.ac.kr, skadry@gmail.com

Abstract

Data aggregation is the significant process in which the information is gathered and combines data to decrease the amount of data transmission in the WSN. The sensor devices are susceptible to node attacks and security issues such as data confidentiality and data privacy are extremely important. A novel technique called Ruzicka Index Regressive Homomorphic Ephemeral Key Benaloh Cryptography (RIRHEKBC) technique is introduced for enhancing the security of data aggregation and data privacy in WSN. By applying the Ruzicka Index Regressive Homomorphic Ephemeral Key Benaloh Cryptography, Ephemeral private and public keys are generated for each sensor node in the network. After the key generation, the sender node performs the encryption using the receiver public key and sends it to the data aggregator. After receiving the encrypted data, the receiver node uses the private key for decrypting the ciphertext. The key matching is performed during the data decryption using Ruzicka Indexive regression function. Once the key is matched, then the receiver collects the original data with higher security. The simulation result proves that the proposed RIRHEKBC technique increases the security of data aggregation and minimizes the packet drop, and delay than the state-of-the-art methods.

Keywords: WSN, Data aggregation, Security and privacy, Homomorphic Ephemeral Key Benaloh Cryptography, Ruzicka Indexive regression

1 Introduction

WSN is a wireless network consists of small sensor nodes and the information gathered by the nodes is to be secured since the attackers hack the data. WSNs are used in different applications such as wild tracking, healthcare, disaster management, military, and so on. However, WSNs are susceptible to a variety of attacks since the distributed nature of the network and directs to higher delays and loss of data aggregation. The conventional system has been implemented for encryption and decryption to overcome

the various security issues. However, it faces several challenges to achieve efficient data aggregation with minimum delay.

An asymmetric key encryption scheme was introduced in [1] based on elliptic curve cryptography to improve the secure end-to-end data aggregation with lesser computation overhead. The designed scheme performance of higher delivery ratio and lesser transmission delay was not minimized. An optimized security model using enhanced fully homomorphic encryption (OSM-EFHE) was introduced in [2] for data aggregation on the large-scale wireless network. The designed model failed to provide privacy-preserving data security in data management.

A Secure and Energy-Efficient Data Aggregation (SEEDA) protocol was developed in [3] using a secret key. However, it failed to focus on preventing more attacks when involving multiple nodes. A Directional Virtual Backbone-based Data Aggregation Scheme (DVBDAS) was introduced in [4] for WVSNs. However, secure data aggregation was not performed. A novel energy-efficient clustering method was introduced in [5] for data aggregation. However, the data aggregation delay was not minimized.

An energy-efficient privacy-preserving data aggregation protocol (EPPA) was designed in [6] to significantly protect the data from attacks and decrease the communication overhead. But the higher throughput of the data aggregation was not achieved. A new lightweight structure-based Data Aggregation Routing (LSDAR) protocol was designed in [7] for

*Corresponding Author: Yunyoung Nam; E-mail: ynam@sch.ac.kr
DOI: 10.53106/160792642021112206008

data protection against malicious threats. The designed routing protocol minimizes the end-to-end delay and packet drop ratio. But the computation overhead was not minimized.

An authorization and verification-based data aggregation method were introduced in [8] to ensure security and minimize the computation overhead. The designed method failed to analyze the multiple attacks during the data aggregation. A symmetric additive homomorphic encryption system was introduced in [9] to offer higher data confidentiality during data collection. But the delay of data aggregation was not minimized.

A Queries Privacy-Preserving mechanism for Data Aggregation (QPPDA) method was developed in [10] to efficiently protect data privacy using a homomorphic encryption scheme. Though the method reduces the computation overhead, the higher packet delivery ratio was not achieved.

The issues identified from the above-said literature are overcome by introducing the novel technique called RIRHEKBC.

The key contribution of the proposed RIRHEKBC technique is listed as given below,

- To increase the security of data aggregation in WSN, a novel machine learning-based cryptosystem called RIRHEKBC is introduced.
- To minimize the computation overhead, the RIRHEKBC generates the pair of Ephemeral keys to perform the encryption and decryption with minimum time consumption. This also helps to reduce the data aggregation time at the sink node.
- To improve packet delivery and minimize packet loss, homomorphic Ephemeral Key Benaloh Cryptography is employed. The Ephemeral key is generated for each session. Once the particular session is completed, then the generated keys are disabled. This helps to avoid the attackers modifying or inject false data. Besides, Ruzicka Indexive regression function is applied for matching the key during the data decryption. As a result, the authorized node obtains the original data from the sender.
- Finally, the simulation is carried out to estimate the performance of the RIRHEKBC technique and other related data aggregation approaches with different metrics. The result discussion shows that the RIRHEKBC technique is highly efficient in secure data aggregation than the other methods.

1.1 Organization of the Paper

The rest of this paper is arranged into five different sections. Section 2, briefly describes the related works. Section 3 describes the RIRHEKBC based secure data aggregation. In Section 4, simulation settings are presented with the different parameters. The performance analysis of the proposed RIRHEKBC technique and other existing methods are presented in Section 5 and finally, the conclusion of the paper is presented in Section 6.

2 Related Works

Data Aggregation (DA) combined with the security method were introduced in [11] for addressing the security issues and achieving higher data confidentiality of data aggregation. A Trust wEighted Secure Data Aggregation algorithm (TESDA) was introduced in [12] to reduce the attacker contribution based on trustworthiness. The designed algorithm increases the attack detection ratio. But the computation overhead was not efficiently minimized.

A firefly algorithm was designed in [13] for aggregating the data with lesser overhead. But the algorithm failed to apply the cryptographic technique to preserve the data from the attackers. The quantum data aggregation was performed in [14] to reduce the overhead using secret sharing and genetic algorithms based on trusted neighbors. But, the attack detection was not performed.

A Distributed Collision-Free Data Aggregation method was developed in [15] to improve the security of data aggregation and minimize the delay. However, the method failed to reduce the computation overhead. The Heavy Weight Security (HWS) algorithm was designed in [16] to ensure the security of data aggregation with minimum overhead. The designed algorithm increases the packet delivery but the higher throughput was not achieved.

A secure data aggregation system was developed in [17] by integrating the homomorphic encryption along with a signature scheme. The designed system reduces the delay but the data delivery was not improved. An efficient security method was introduced in [18] for data aggregation to improve data integrity and confidentiality. But the scheme failed to apply the efficient cryptography mechanism for achieving higher confidentiality as well as data privacy.

A trust Assisted- Energy Efficient Aggregation (TA-EEA) system was introduced in [19] to increase the performance of aggregation. But the system failed to support large-scale Wireless Networks. An additively homomorphic encryption and fragmentation method (AHEF) was introduced in [20] for data aggregation with lesser overhead. But the higher delivery ratio was not achieved during the data aggregation.

3 Proposal Methodology

A WSN is a heterogeneous network that includes hundreds of thousands of low-cost and low-power small sensor nodes to sense and collect the information from the deployment environment. Data aggregation is the process of collecting useful data from the sensor

nodes. During the data aggregation of WSN, data aggregation is a challenging task to protect sensitive information transmitted by sensor nodes in wireless networks from passive attacks. Therefore, security is a major concern in wireless sensor networks.

3.1 System Model

In this section, the system model of the proposed technique RIRHEKBC is presented. A WSN is represented by a graphical model 'G = (V, E)' in which 'V' denotes a sensor node SD = {sd1, sd2, …, sdn} and 'E' represents edges i.e., links between the sensor nodes.

Figure 1 illustrates the data aggregation process where a set of sensor nodes SD = {sd_1, sd_2, …, sd_n} and it is responsible for sensing and gathering the information from an environment and send it to an aggregator and forward the base station

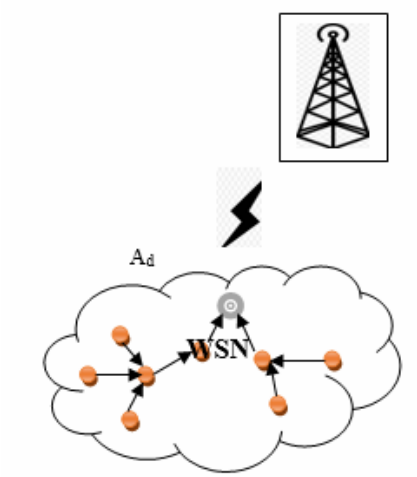

Figure 1. Data aggregation process in a WSN

As shown in Figure 1, the aggregator node (A_d) acts as a data collector to collect the data from the other reliable sensor nodes in the network. During the data transmission, the distributed sensor nodes transmit the sensed data d_1, d_2, …, d_m to the aggregator node in a secured manner.

A novel RIRHEKBC technique is introduced for secure data transmission between sender and receiver. Security of the network is assured by an authorized node to aggregate the sensed data from the sensor node by avoiding various attacks. Based on the motivation, the RIRHEKBC technique uses the Homomorphic Ephemeral key Benaloh Cryptographic function for secure data aggregation in WSN. The different process of proposed RIRHEKBC technique is described in the following sections.

Benaloh Cryptography is public-key cryptography that means the designed cryptographic system uses the pairs of Ephemeral keys such as private and public during the data transmission [21-24]. Ephemeral key i.e., both private and the public key is then generated for each execution of a key establishment process. The private keys are only known to the data owner and the public keys are distributed widely in the network. The Benaloh Cryptographic technique consists of four different processes namely Ephemeral key generation, data encryption, key matching, and decryption. Figure 2 shows the flow process of the Homomorphic Ephemeral key Benaloh Cryptographic function to improve the confidentiality and privacy in WSN.

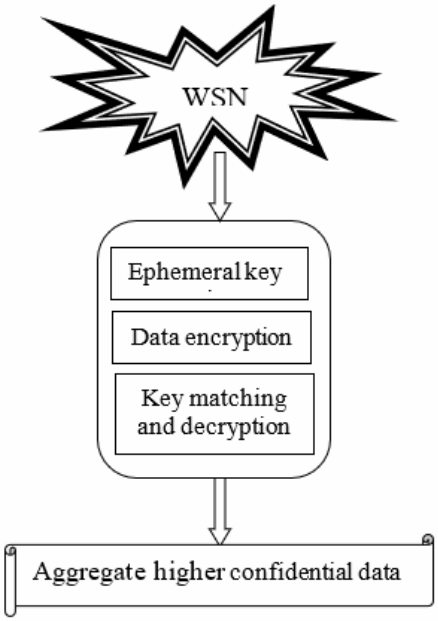

Figure 2. Flow process of Homomorphic Ephemeral key Benaloh Cryptographic function

Initially, the Ephemeral pair of keys are generated for each sensor node. The generated keys are used for both encryptions as well as decryption. In other words, a different pair of keys is generated for each session. It helps to improve the security of data transmission between the sender and receiver. In the key generation process, the public and private keys are generated for each sensor node that participated in the data transmission process [21]. The conventional cryptographic systems use the symmetric-key algorithm and it uses the common shared secret key. It may be hacked by the attackers and it also degrades the confidentiality level of data. Therefore, a proposed RIRHEKBC technique generates an Ephemeral key for each session

Once the particular session is finished, then the generated keys are disabled. A session is a sequence of interactions between the two communications. Then the algorithm creates a new Ephemeral key for the next session. This helps to avoid unauthorized access (i.e., Data Injection Attack, Compromised Node Attack, replication attack, wormhole attack) from the cloud server and also improves the confidentiality level. A data injection attack is a malicious code introduced in the network which gets sensed information from the sensor node to the attacker. A compromised node creates the false data and it is injected into the WSN data aggregation and it also drops the aggregation

information.

By introducing a wormhole attack, an attacker records the information's at a particular location within the network and, tunnels them to another location. The node replication attack is also called a replica attack in WSN. The adversary captures the deployed sensor nodes from the network and extracts all the significant information and functions to generate duplication nodes. The generated duplication nodes also have information such as the ID of the captured sensor node. Then, the adversary distributes the generated duplication nodes in the network. It interrupts the network transmission and also disturbs the entire network operation and also introduces false data and destroys the operation of the normal sensor node operation such as data aggregation, and so on. Therefore, the proposed RIRHEKBC technique solves these kinds of attacks to improve security.

3.2 Ephemeral Key Generation

Key generation is the initial process of cryptography and the generated keys are used to perform the encryption and decryption [25]. Let us consider and select two prime numbers p and q in a random manner.

Therefore, the key generation process of the cryptography technique is given below,

$$R = p * q \tag{1}$$

$$Q = (p-1)(q-1) \tag{2}$$

$$\beta = \alpha^{\frac{Q}{s}} \bmod R \tag{3}$$

From Eq. (1) (2) (3), (α, R) denotes a public key, (Q, β) represents the private key, s indicates a block size, α indicates an integer number. After the key generation, the public key is distributed through the network and the private key is kept secret.

3.3 Encryption

After the key generation, the proposed RIRHEKBC technique performs the encryption at the sender side to encrypt the original data i.e., plaintext into a ciphertext for avoiding the attacker node access. The sensor node performs the encryption using the receiver public key. Here, the sender node is a sensor node and the receiver node is a data aggregator.

Figure 3 depicts the block diagram of the encryption process to improve the security of data transmission from sensor nodes to aggregators. Let us consider the data sensed to be 'd'. The input data is encoded into a string of bits ($b_i = b_1, b_2, \ldots, b_m$). The ciphertext of the input data is generated as given below,

$$a = \alpha^b r^s \bmod R \tag{4}$$

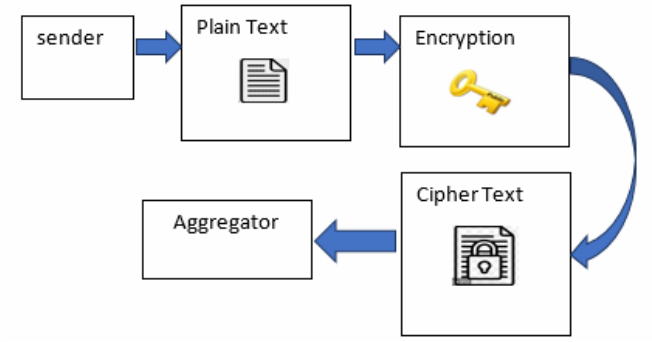

Figure 3. Block diagram of the encryption process

Where **c** indicates a ciphertext, **b** represents a message bit, **α** denotes a public key, **r** indicates a random number, **s** represents a block size, **R** indicates a product of two prime numbers. Then the sender node sends the encrypted ciphertext into the data aggregator.

3.4 Ruzicka Indexive Regressed Key Matching and Decryption

The authorized node enters the private key for decrypting the ciphertext and it is verified by the base station to collect the sensed data from the sender node. On the contrary to the conventional cryptographic technique, the proposed algorithm uses the Ruzicka Indexive regression to perform key matching in the data decryption process. Regression is a machine learning technique to measure the similarity between the two variables. Here the regression is used to find the similarity between the entered private key at the time of decryption and the already generated private key at the time of key generation. The similarity is measured using the Ruzicka index function. The mathematical formula for calculating the similarity between the keys are given below,

$$\delta = \frac{PK_E \bigcap PK_G}{\Sigma PK_E + \Sigma PK_G - PK_E \bigcap PK_G} \tag{5}$$

Where 'δ' represents a similarity coefficient, PK_E represents entered the private key, PK_G denotes a generated private key. $PK_E \cap PK_G$ denotes a mutual dependence between the two keys.

The Ruzicka similarity coefficient (δ) provides the similarity value between 0 and 1.

$$\delta = \{1, matched\ 0, not\ matched\} \tag{6}$$

The Ruzicka similarity coefficient returns '1' that indicates that both the keys are correctly matched and then the receiver is said to an authorized. Otherwise, the keys are not matched, and the receiver node is said to an attacker. Once the key is matched, decryption is performed to decrypt the ciphertext and obtain the original data.

Figure 4 illustrates the block diagram of the key matching and decryption process to obtain the original data at the receiver (i.e., aggregator). Once the key is matched, original plaintext is obtained as follows,

$$d = (H) \quad \textbf{(7)}$$

$$H = C^{\frac{Q}{s}} \bmod R \quad \textbf{(8)}$$

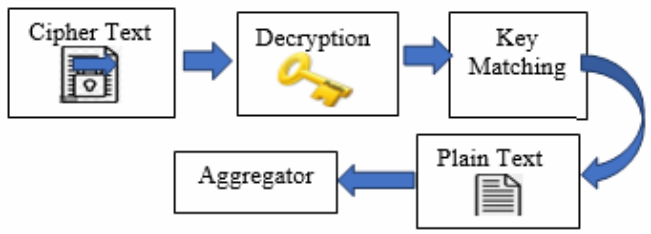

Figure 4. Block diagram of the Ruzicka Indexive regressed key matching and decryption

Where **d** denotes an original text, **Q** indicates a private key. As a result, the aggregator receives the original sensed data by avoiding the attacker node. This helps to enhance the security of data aggregation as well as increase data confidentiality. The packet delivery ratio is increased using the Homomorphic Ephemeral Key Benaloh Cryptography in the RIRHEKBC technique. The Homomorphic Ephemeral Key Benaloh Cryptography is used to deliver the data packets to the aggregator node. The aggregator node gathers the sensed data from the sensor node. The encryption, decryption process is performed for avoiding attacker node access. Followed by, this helps to avoid unauthorized access from the cloud server and correctly transmits the data delivery from sender to receiver. As a result, the proposed RIRHEKBC technique improves the packet delivery ratio and minimizes the packet drop due to the attacker in the network. The flow chart of the proposed RIRHEKBC technique is given in Figure 5 below. At first, the number of sensor nodes is deployed in the heterogeneous network to sense and send the information to the data aggregator. The private and public keys are created for each sensor node.

Then, the created keys are employed to carry the encryption and decryption. Next, the entered key is matched with the created key with aid of the Ruzicka similarity index. When both keys are matched, then the receiver is the authorized node. Next, decryption is performed. Otherwise, Keys are not matched and the receiver node is said to an attacker. This achieves the higher secure data aggregation in WSN. The algorithmic process of the RIRHEKBC technique is described as given below,

Algorithm 1 describes the step-by-step process of secure data aggregation using Ruzicka Index Regressive Homomorphic Ephemeral Key Benaloh Cryptography in the WSN. Initially, the sensor nodes are deployed in the network for sensing the data in the network. For each sensor node, the private and public keys are generated for secure data aggregation. After the key generation, the sensor node starts to transmit

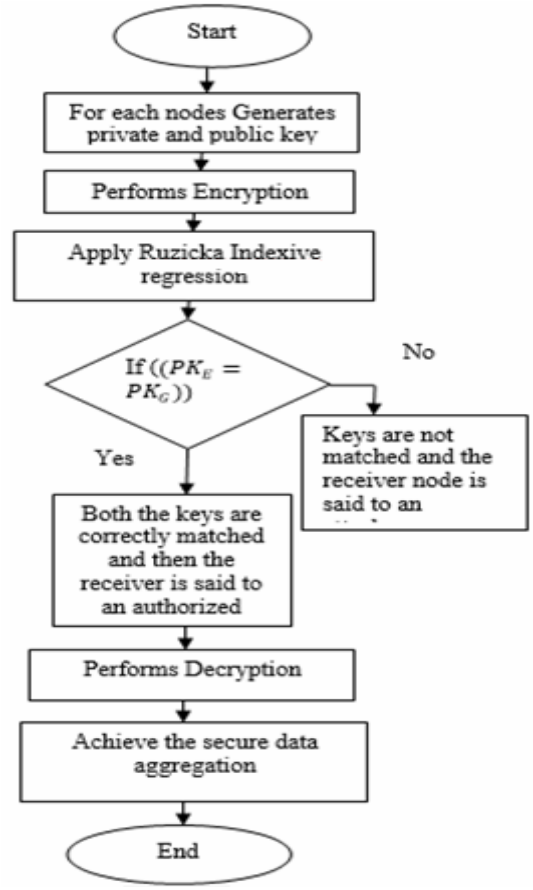

Figure 5. Flowchart of proposed RIRHEKBC technique

Algorithm 1. Ruzicka Index Regressive Homomorphic Ephemeral Key Benaloh Cryptography

Input: Number of sensor nodes $SD = \{Sd_1, Sd_2, \ldots, Sd_n\}$, Number of data $d_1, d_2, d_3, \ldots, d_m$

Output: Improve the secure data aggregation

Begin

Step 1: **for each 'node**

Step 2: **G**enerates private and public key

Step 3: **For** each sensed data 'd_i'

Step 4: Encode data into a bit string $b_i = b_1, b_2, b_b$

Step 5: Convert plain text into ciphertext

$$c = \alpha^b r^s \bmod R$$

Step 6: Send ciphertext into the receiver

Step 7: **End for**

Step 8: **Receiver enters the private for decryption**

Step 9: **if** $(PK_E = PK_G)$ **then**

Step 10: δ returns '1'

Step 11: The receiver is said to be an authorized

Step 12: **else**

Step 13: δ returns '0'

Step 14: The receiver is said to be an attack

Step 15: **end if**

Step 16: Authorized receiver obtain the original data 'd'

Step 17: **end for**

End

the collected data into an aggregator. Before transmission, the sender node encrypts the sensed data and it is transferred to the receiver node. On the receiver side, the decryption is performed using the receiver's private key. The entered key is verified with the generated key using the Ruzicka similarity index. If these two keys are matched, then the receiver is said to be an authorized node and it receives the original text. Otherwise, the decryption is not performed. This helps to improve the secure data aggregation in WSN.

4 Simulation Setup

In this section, the performance of the proposed RIRHEKBC technique and three existing methods namely asymmetric key encryption scheme [1], OSM-EFHE [2], EPPA [6] are implemented in the NS2 network simulator. In order to perform secure data aggregation in WSN, 500 sensor nodes are randomly deployed over a squared area of A^2 (1100 m * 1100 m). In the simulation system, a random waypoint model is used as a mobility model to perform secure data aggregation in WSN. The Ad hoc on-demand distance vector routing protocol (AODV) is applied to detect the different types of attacks namely Data Injection Attack, Compromised Node Attack, replication attack, wormhole attack in the simulation scenario. The simulation time is set as 300 seconds and the sensor nodes' speed is varied from 0-20m/s. The various parameters are listed in Table 1.

Table 1. Simulation parameters

Simulation Parameters	Values
Simulator	NS2
Network area	1100 m * 1100 m
Number of sensor nodes	50-500
Mobility model	Random Waypoint
Number of data packets	25-250
Speed of node	0 – 20 m/s
Simulation time	300s
Number of runs	10
Protocol	AODV

5 Results and Discussion

The simulation results of the four different methods namely the proposed RIRHEKBC technique, asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6] are discussed with respect to different metrics such as computation overhead, packet delivery ratio, packet drop rate, delay, and throughput. This metrics assessment is done with the help of a table and graphical representation.

5.1 Computation Overhead

Computation overhead is measured as the amount of time taken to perform data encryption, data decryption, and data aggregation. Therefore, the formula for calculating the overall computation overhead is given below,

$$CO = [E_t + D_t + A_t] \quad \textbf{(9)}$$

Where CO denotes a computation overhead, E_t indicates the encryption time, D_t denotes a decryption time, A_t indicates the aggregation time. Computation overhead is measured in terms of seconds (sec).

Table 2 demonstrates the performance analysis of computation overhead using four different methods namely RIRHEKBC technique, asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6]. As shown in Table 2, ten different runs are carried out for each method with different simulation time ranges from 10to 100sec.

Table 2. Comparison of computation overheads

Simulation time (sec)	Computation overhead (sec)			
	RIRHEKBC	asymmetric key encryption scheme	OSM-EFHE	EPPA
100	0.3	0.45	0.68	0.72
200	0.63	0.68	0.75	0.89
300	0.68	0.76	0.81	0.9
400	0.72	0.8	0.88	1.1
500	0.85	0.92	0.99	1.4
600	0.92	1.22	1.56	1.76
700	1.3	1.54	1.7	1.9
800	1.6	1.9	1.97	2.2
900	1.7	2.1	2.5	2.7
1000	1.9	2.4	2.7	2.7

From the observed results, it is noticeable that the proposed RIRHEKBC technique minimizes the computation overhead. Let us set it to 100 sec for simulation time to conduct the experiments. By applying the RIRHEKBC technique, the overhead that occurred during the encryption, decryption, and data aggregation is 0.3sec. Whereas, the computation overhead of the asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6] are consumed by 0.45 sec, 0.68 sec, and 0.72 sec respectively. The average of ten results indicates that the computation overhead is significantly minimized by 16%, 27%, and 37% when compared to the [1-2, 6] respectively.

Figure 6 portrays the performance results of a computation overhead using four different techniques according to the different simulation times. The simulation time is taken in the horizontal axis and the computation overhead is obtained at the vertical axis. Four different colors of columns blue, red, green, and violet indicate the computation overhead. From Figure 6, it is noticed that the computation overhead is said to be minimized using the RIRHEKBC technique when compared to conventional techniques. This is due to the application of the Ephemeral Key Benaloh Cryptography technique. The proposed technique

consumes lesser time for data encryption and data decryption resulting in reduces time of data aggregation.

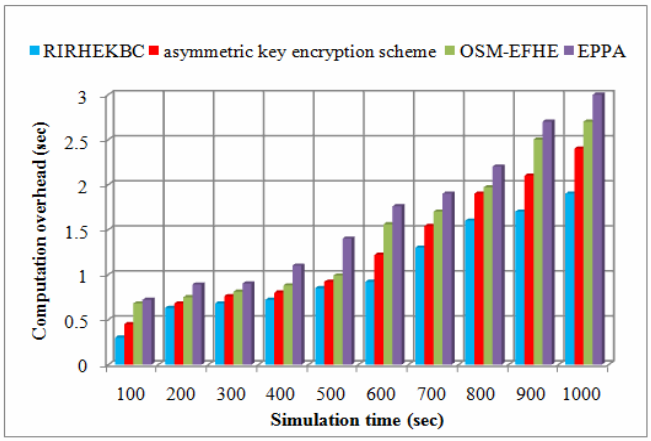

Figure 6. Graphical illustration of the computation overheads

5.2 Packet Delivery Ratio

Packet delivery ratio is defined as the ratio of the number of data packets correctly received at the aggregator node to the total number of data packets sent from the sender node at different simulation times. The Packet Delivery Ratio (PDR)is mathematically calculated as given below,

$$PDR = \left[\frac{Number\ of\ data\ packets\ received}{Number\ of\ data\ packets} \right] * 100 \quad \textbf{(10)}$$

Where PDR indicates a Packet Delivery Ratio and it is measured in terms of percentage (%).

Table 3 illustrates the simulation results of packet delivery ratio along with the different simulation times using four methods namely RIRHEKBC technique, asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6]. From the observed results, the RIRHEKBC technique achieves a higher packet delivery ratio than the other three methods. Let us consider the sample mathematical calculation with the simulation time of 10sec and the number of data packets considered for experimentation is 100. By applying the RIRHEKBC technique, 92 packets are successfully received at the data aggregator and the delivery ratio is 92% and the delivery ratio of existing [1-2, 6] are 88%, 84%, and 80% respectively. Similarly, the experiment is conducted with the different simulation times and similar data being sent from source to aggregator node. The average of ten results indicates that the packet delivery ratio of the RIRHEKBC technique is comparatively increased by 6% when compared to [1], 10% when compared to [2], and 15% when compared to [6].

Table 3. Comparison of packet delivery ratio

Simulation time (sec)	Packet Delivery Ratio (%)			
	RIRHEKBC	asymmetric key encryption scheme	OSM-EFHE	EPPA
10	92	88	84	80
20	93	89	85	81
30	92	86	83	78
40	91	85	82	80
50	92	86	83	81
60	93	87	84	81
70	92	86	83	79
80	94	88	84	82
90	94	89	85	84
100	93	87	84	81

Figure 7 portrays the comparative results of a packet delivery ratio using four different techniques with respect to the different simulation times. The number of data packets is given to the different simulation times and the corresponding packet delivery ratio is obtained at the vertical axis. The observed results inferred that the packet delivery ratio is said to be minimized using the RIRHEKBC technique when compared to existing techniques. The Homomorphic Ephemeral Key Benaloh Cryptography is applied for delivering the data packets to the aggregator node. The aggregator node collects the sensed data from the sensor node.

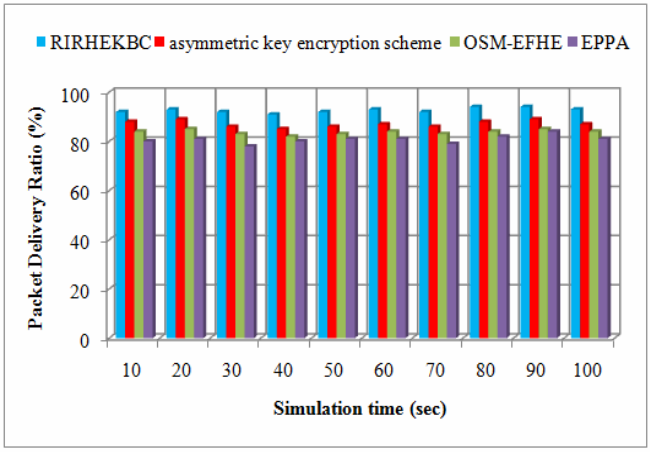

Figure 7. Graphical illustration of the packet delivery ratio

5.3 Performance Analysis of Packet Drop Rate

The packet drop rate is defined as the ratio of the size of data packets dropped during the data aggregation. The Packet drop rate is calculated as given below,

$$R_{PD} = \left[\frac{data\ packets\ dropped\ (bytes)}{data\ packets\ size\ (bytes)} \right] * 100 \quad \textbf{(11)}$$

Where R_{PD} indicates a packet drop rate and it is measured in terms of percentage (%).

Table 4 and Figure 8 indicate the comparative analysis of a packet drop rate using four different techniques. As shown in Figure 8, the various results of drop rates are observed for different simulation times. The observed results inferred that the packet drop rate is found to be minimized by using the RIRHEKBC technique when compared to existing techniques. This is due to the application of secure data aggregation by avoiding the four types of attacks using the Ruzicka Indexive regression function. During the decryption process, the receiver node enters the private key for decryption. The Ruzicka Indexive regression verifies the entered key is matched with the generated key. If these keys are matched, then the receiver is said to be authorized and the get the original data. This helps to minimize the packet drop due to the attacks. The simulation is conducted with 100 bytes of the data being transmitted from the sender node. By applying the RIRHEKBC technique, 42.5 bytes of data are dropped whereas the 63.5bytes of data, 68.8bytes of data and 75.3bytes of data are dropped using asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6]. Similarly, the various simulation time is set to obtain the various results. The observed results indicate that the RIRHEKBC technique minimizes the packet drop rate by 33%, 43%, and 48% when compared to [1-2, 6] respectively.

Table 4. Comparison of packet drop rate

Simulation time (sec)	Packet drop rate (%)			
	RIRHEKBC	asymmetric key encryption scheme	OSM-EFHE	EPPA
10	42.5	63.5	68.8	75.3
20	40.3	58.6	65.3	71.6
30	38.6	56.5	63.2	69.8
40	35.3	54.2	60.1	67.4
50	32.2	50.5	58.5	65.5
60	31.1	48.5	56.8	62.3
70	30.5	46.2	55.1	60.2
80	29.3	43.5	54.5	59.4
90	28.2	40.7	53.6	57.6
100	26.5	38.2	51.5	55.8

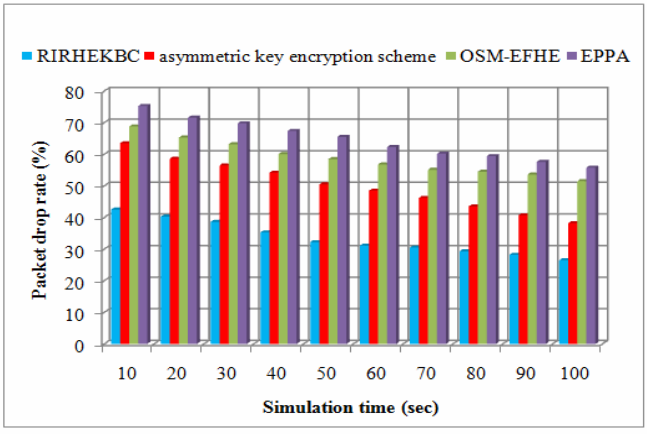

Figure 8. Graphical illustration of the packet drop rate

5.4 Performance Analysis of Transmission Delay

Transmission delay is measured as the difference between the actual arrival time of the data packets and the observed arrival time of the data packets at the aggregator node. The overall transmission delay is mathematically calculated as given below,

$$TD = [t_{act}] - [t_{obs}] \quad \textbf{(12)}$$

Where TD denotes a transmission delay, t_{act} denotes an actual arrival time and t_{obs} indicates an observed arrival time. The overall transmission delay is measured in terms of seconds (sec).

Table 5 describes the simulation results of transmission delay based on sizes of data packets being sent from the sender node with three different methods namely the proposed RIRHEKBC technique, asymmetric key encryption scheme [1], OSM-EFHE [2], and EPPA [6]. Among three methods, the RIRHEKBC technique effectively minimizes the delay of data packet transmission between sender and receiver.

Table 5. Comparison of transmission delay

Packet size (bytes)	Transmission delay (sec)			
	RIRHEKBC	asymmetric key encryption scheme	OSM-EFHE	EPPA
100	.08	1.5	1.8	2.1
200	1.1	1.7	2.2	2.4
300	1.2	1.8	2.4	2.7
400	1.4	2.1	2.6	2.9
500	1.5	2.3	2.8	3.2
600	1.7	2.5	3.1	3.4
700	1.8	2.7	3.3	3.6
800	2.1	2.8	3.5	3.7
900	2.2	3.1	3.6	3.9
1000	2.3	3.2	3.8	4

The convergence graph of transmission delay is shown in Figure 9. For 100 bytes sizes of data packets, the delay of the RIRHEKBC technique is 1.08s and the transmission delay of existing [1-2, 6] is 1.5s, 1.8s, and 2.1s respectively. The average of ten results indicates that the transmission delay of the RIRHEKBC technique is minimized by 31%, 44%, and 49% when compared to the existing methods. The reason is because of the application of the Homomorphic Ephemeral key Benaloh Cryptographic function applied in the proposed RIRHEKBC technique to secure data aggregation in WSN. An efficient cryptosystem for securely transmit the data from a sender node to an aggregator node. The aggregator node groups the sensed data from the sensor node to avoid dissimilar attacks namely Data Injection Attack, Compromised Node Attack, replication attack, wormhole attack. This helps to minimize the transmission delay.

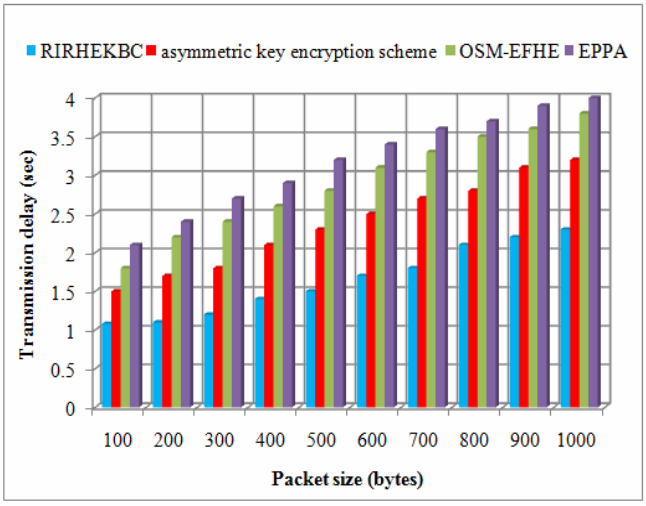

Figure 9. Graphical illustration of transmission delay

5.5 Performance Analysis of Throughput

Throughput is measured as the amount of data packets successfully delivered at an aggregator node in a given period. The throughput is mathematically formulated as given below,

$$T = \left[\frac{Size\ of\ dapa\ packet\ received\ (bits)}{time\ (sec)} \right] \quad \textbf{(13)}$$

From the above mathematical expression (13), 'T' symbolizes the measured in terms of bits per second (bps).

Table 6 illustrates the performance results of the throughput based on the size of data packets that varies in the ranges from 100-1000 bytes. From the observed results, the proposed RIRHEKBC technique delivers a large amount of data packets within a particular second when compared to the conventional schemes. This is proved through the sample calculation. Let us consider the 100 bytes sizes of data packets being sent from the sender node. By applying the proposed RIRHEKBC technique, 60 bits of the data packets are delivered at the aggregator node. Similarly, by applying the [1-2, 6] 50 bits, 43 bits, and 38 bits of data packets are successfully delivered in one second at the destination. Followed by, the various performance results are observed for each method. The obtained results of the proposed RIRHEKBC technique are compared to the existing results. The average of comparison results indicates that the throughput of the RIRHEKBC technique is increased by 11% when compared to asymmetric key encryption scheme [1], 20% when compared to OSM-EFHE [2], and 29% when compared to EPPA [6].

Figure 10 depicts the performance analysis of throughput according to different sizes of the data packets 100 bytes-1000 bytes. As illustrated in Figure 10, the results of throughput using three techniques RIRHEKBC technique, asymmetric key encryption

Table 6. Comparison of throughput

Packet size (bytes)	Throughput (bps)			
	RIRHEKBC	asymmetric key encryption scheme	OSM-EFHE	EPPA
100	60	50	43	38
200	109	89	78	73
300	218	205	193	182
400	346	300	280	250
500	432	410	387	365
600	574	534	510	480
700	655	620	580	540
800	768	720	680	630
900	932	812	780	755
1000	1013	923	879	845

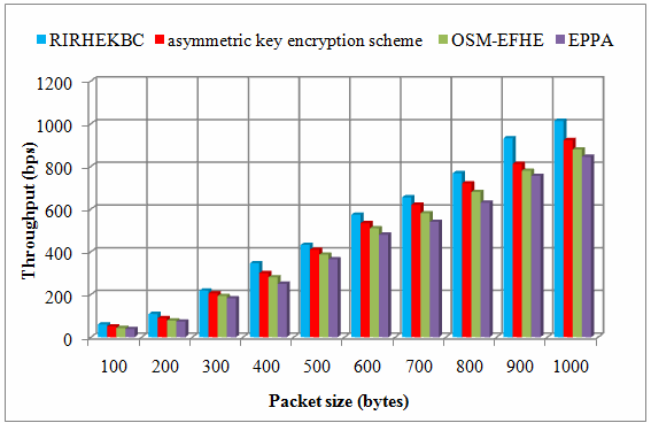

Figure 10. Graphical illustration of throughput

scheme [1], OSM-EFHE [2], EPPA [6] are represented by a variety of colors namely green, red, blue, and violet color respectively. The plot indicates that the proposed RIRHEKBC technique provides superior performance in terms of achieving higher throughput. This is because of the Homomorphic Ephemeral key Benaloh Cryptographic function used in the proposed RIRHEKBC technique for secure data aggregation in WSN. The proposed RIRHEKBC technique securely transmits the sensed data packets to the data aggregator. The aggregator node gathers the sensed data from the sensor node for avoiding the different attacks (i.e., Data Injection Attack, Compromised Node Attack, replication attack, wormhole attack). Moreover, the packet drop of the data is also minimized due to the attacks in the distributed network.

6 Conclusion

In WSN, it is critical to guarantee secure data aggregation to increase the packet delivery ratio with a minimum overhead as well as delay. In this paper, the RIRHEKBC technique is introduced for secure data aggregation between sensors and the sink node. The proposed cryptosystem generates a pair of ephemeral keys for secure transmission in WSN. The sender node encrypts the data with the public key of the receiver and sent. On the receiver side, the private key is used

to decrypt the original data. Before getting the original data, the Ruzicka Index Regression function is applied for matching the private key. This helps to minimize unauthorized attacks such as Data Injection attacks, Compromised Node attacks, replication attacks, and wormhole attacks. The simulation is conducted with the different performance metrics such as the computation overhead, packet delivery ratio, packet drop rate, transmission delay, and throughput with respect to a number of data packets and sizes. The observed results proved that the RIRHEKBC technique improves the security of data aggregation WSN with a higher delivery ratio and minimum packet drop as well as delay than the state-of-the-art methods.

Acknowledgements

This research was supported by Korea Institute for Advancement of Technology(KIAT) grant funded by the Korea Government(MOTIE) (P0012724, The Competency Development Program for Industry Specialist) and the Soonchunhyang University Research Fund.

References

[1] X. Qi, X. Liu, J. Yu, Q. Zhang, A Privacy Data Aggregation Scheme for Wireless Sensor Networks, *Procedia Computer Science*, Vol. 174, pp. 578-583, 2020.

[2] M. Shobana, R. Sabitha, S. Karthik, An enhanced soft computing-based formulation for secure data aggregation and efficient data processing in large-scale wireless sensor network, *Soft Computing*, Vol. 24, No. 16, pp. 12541-12552, August, 2020.

[3] A. A. Jasim, M. Y. I. B. Idris, S. R. B. Azzuhri, N. R. Issa, N. B. M. Noor, J. Kakarla, I. S. Amiri, Secure and Energy-Efficient Data Aggregation Method Based on an Access Control Model, *IEEE Access*, Vol. 7, pp. 164327-164343, November, 2019.

[4] J. Zhang, S. Liu, P. Tsai, F. Zou, X. Ji, Directional virtual backbone based data aggregation scheme for Wireless Visual Sensor Networks, *PLoS One*, Vol. 13, No. 5, pp. 1-27, May, 2018.

[5] S. G. Shah, A. Ahmed, I. Ullah, W. Noor, A novel data aggregation scheme for wireless sensor networks, *International Journal of Advanced Computer Science and Applications*, Vol. 10, No. 2, pp. 585-590, 2019.

[6] X. Liu, J. Yu, X. Zhang, Q. Zhang, C. Fu, Energy-efficient privacy-preserving data aggregation protocols based on slicing, *Journal on Wireless Communications and Networking*, Vol. 2020, Article No. 19, January, 2020.

[7] K. Haseeb, N. Islam, T. Saba, A. Rehman, Z. Mehmood, LSDAR: A light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks, *Sustainable Cities and Society*, Vol. 54, pp. 1-9, March, 2020.

[8] S. N. Nels, J. A. P. Singh, Security-aware authorization and verification based data aggregation model for wireless sensor networks, *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, Vol. 34, No. 3, pp. 1-20, May-June, 2021.

[9] V. S. Lakshmi, P. P. Deepthi, A secure channel code-based scheme for privacy preserving data aggregation in wireless sensor networks, *International Journal of Communication Systems*, Vol. 32, No. 1, pp. 1-21, January, 2019.

[10] X. Liu, X. Zhang, J. Yu, C. Fu, query privacy preserving for data aggregation in wireless sensor networks, *Wireless Communications and Mobile Computing*, Vol. 2020, pp. 1-10, February, 2020,

[11] X. Liu, J. Yu, F. Li, W. Lv, Y. Wang, X. Cheng, Data aggregation in wireless sensor networks: from the perspective of security, *IEEE Internet of Things Journal*, Vol. 7, No. 7, pp. 6495-6513, July, 2020.

[12] P. Padmaja, G. V. Marutheswar, Energy efficient data aggregation in wireless sensor networks, *Materials Today: Proceedings*, Vol. 5, No. 1, pp. 388-396, 2018.

[13] I. Mosavvar, A. Ghaffari, Data aggregation in wireless sensor networks using firefly algorithm, *Wireless Personal Communications*, Vol. 104, No. 1, pp. 307-324, January, 2019

[14] T. H. Kim, S. Madhavi, Quantum data aggregation using secret sharing and genetic algorithm, *IEEE Access*, Vol. 8, pp. 175765-175775, September, 2020.

[15] D. Qin, Y. Zhang, J. Ma, P. Ji, P. Feng, A distributed collision-free data aggregation scheme for wireless sensor network, *International Journal of Distributed Sensor Networks*, Vol. 14, No. 8, pp. 1-15, August, 2018.

[16] A. Saravanaselvan, B. Paramasivan, Design and implementation of an efficient attack resilient computation algorithm in WSN nodes, *Cluster Computing*, Vol. 22, No. Supplement 2, pp. 3301-3311, March, 2019.

[17] H. Zhong, L. Shao, J. Cui, Y. Xu, An efficient and secure recoverable data aggregation scheme for heterogeneous wireless sensor networks, *Journal of Parallel and Distributed Computing*, Vol. 111, pp. 1-12, January, 2018.

[18] O. O. Olakanmi, K. O. Odeyemi, A secure and collaborative data aggregation scheme for fine-grained data distribution and management in Internet of Things, *Security and Privacy*, Vol. 4, No. 1, pp. 1-17, January-February, 2021.

[19] A. Latha, S. Prasanna, S. Hemalatha, B. Sivakumar, A harmonized trust assisted energy efficient data aggregation scheme for distributed sensor networks, *Cognitive Systems Research*, Vol. 56, pp. 14-22, August, 2019.

[20] K. Kapusta, G. Memmi, H. Noura, Additively homomorphic encryption and fragmentation scheme for data aggregation inside unattended wireless sensor networks, *Annals of Telecommunications*, Vol. 74, No. 3-4, pp. 157-165, April, 2019.

[21] L. Krishnasamy, R. K. Dhanaraj, D. G. Gopal, T. R. Gadekallu, M. K. Aboudaif, E. A. Nasr, A heuristic angular clustering framework for secured statistical data aggregation in sensor networks, *Sensors*, Vol. 20, No. 17, Article No.

4937, September, 2020.

[22] S. Gomathi, C. G. Krishnan, Malicious node detection in wireless sensor networks using an efficient secure data aggregation protocol, *Wireless Personal Communications*, Vol. 113, No. 4, pp. 1775-1790, August, 2020.

[23] W. Fang, X. Wen, J. Xu, J. Z. Zhu, CSDA: a novel cluster-based secure data aggregation scheme for WSNs, *Cluster Computing*, Vol. 22, No. Supplement 3, pp. 5233-5244, May, 2019

[24] S. Agrawal, M. L. Das, J. Lopez, detection of node capture attack in wireless sensor networks, *IEEE Systems Journal*, Vol. 13, No. 1, pp. 238-247, March, 2019.

[25] R. Hajian, S. H. Erfani, CHESDA: continuous hybrid and energy-efficient secure data aggregation for WSN, *The Journal of Supercomputing*, Vol. 77, No. 5, pp. 5045-5075, May, 2021.

Biographies

Saravanakumar Pichumani has finished his Bachelor of Engineering in Electronics and Communication Engineering from Bharathiyar University in 2001 and pursued his Master's degree in Information Technology from College of Engineering, Anna University in 2003. He has almost 15 years of Teaching Experience and 2 years of Industrial Experience.

T. V. P. Sundararajan received his B.E. degree in Electronics and Communication Engineering from Kongu Engineering College, Bharathiyar University, Coimbatore, Tamil Nadu (1993). He received his M.E. Degree in Applied Electronics from GCT, Bharathiyar University, Coimbatore, Tamil Nadu (1999). He received his Ph.D. degree from Anna University, Chennai (2013) for his work in Mobile Ad hoc Network Security.

Rajesh Kumar Dhanaraj is an Associate Professor in the School of Computing Science and Engineering at Galgotias University, Greater Noida, India. He holds a PhD degree in Information and Communication Engineering from the Anna University Chennai, India. He has contributed 20+ books on various technologies and 35+ articles.

Yunyoung Nam received the B.S., M.S., and Ph.D. degrees in computer engineering from Ajou University, Korea in 2001, 2003, and 2007 respectively. He was a Senior Researcher with the Center of Excellence in Ubiquitous System, Stony Brook University, Stony Brook, NY, USA, from 2007 to 2010, where he was a Postdoctoral Researcher, from 2009 to 2013. He was a Research Professor with Ajou University, from 2010 to 2011. He was a Postdoctoral Fellow with the Worcester Polytechnic Institute, Worcester, MA, USA, from 2013 to 2014. He was the Director of the ICT Convergence Rehabilitation Engineering Research Center, Soonchunhyang University, from 2017 to 2020. He has been the Director of the ICT Convergence Research Center, Soonchunhyang University, since 2020, where he is currently an Assistant Professor with the Department of Computer Science and Engineering. His research interests include multimedia database, ubiquitous computing, image processing, pattern recognition, context-awareness, conflict resolution, wearable computing, intelligent video surveillance, cloud computing, biomedical signal processing, rehabilitation, and healthcare systems.

Seifedine Kadry has a Bachelor degree in 1999 from Lebanese University, MS degree in 2002 from Reims University (France) and EPFL (Lausanne), PhD in 2007 from Blaise Pascal University (France), HDR degree in 2017 from Rouen University. At present his research focuses on Data Science, education using technology, system prognostics, stochastic systems, and applied mathematics. He is an ABET program evaluator for computing, and ABET program evaluator for Engineering Tech.