Received: August 15, 2022. Revised: October 28, 2022.

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Masked Location based Key Exchange Mechanism for IoT Nodes

Anandhavalli A[1]* **Bhuvaneswari A[1]**

[1]*Department of Computer Science, Cauvery College for Women (Autonomous), Affiliated to Bharathidasan University, Tamil Nadu, India*
* Corresponding author's Email: anandhavalli.ca@cauverycollege.ac.in

Abstract: Internet-of-Things (IoT) is one of the modern communication frameworks which is used to interconnect heterogeneous computational devices and to provide the expertise to exchange data among them without human-to-human or human-to-computer interdependence. This versatile heterogeneous communication is achieved by the unique identifier abstraction. Key exchange mechanisms are playing a vital role in IoT to procure the secured communication between the devices. The challenge in designing Key Exchange Mechanisms for IoT is increased due to the heterogeneous nature of the system. The IoT network can have some high-power computational rigs along with tiny low powered domestic appliances at the same time. The key exchange mechanism should be secure enough to protect the communication between the high computational power nodes and it should also be low power operational to deal with the low computational power devices. The proposed work of MLKEM formed the new Key Exchange Mechanism for IoT nodes based on Dual Rosernberg Pairing Location Masker and Fuzzy Miller's Elliptic Curve. The result of this proposed work is to exchange the new authentication key between the IoT nodes with secured manner. The new Key Exchange Mechanism are analysed by OPNET Network Simulator Tool. The introduction of Dual Rosenberg Pairing Location Mask is used to improve the security and Fuzzy Miller's Elliptic Curve Key exchange is used to provide a power rational communication between the nodes in the IoT network.

Keywords: Elliptic curve key exchange, Internet-of-things (IoT), Key exchange mechanism, Network security, Lightweight security, Rosenberg pairing.

1. Introduction

Modern world human league is entangled with the smart gadgets predominantly. From fundamental person- to- person communication to healthcare emergency administrations are handled by the virtue of intelligent gadgets around the people. These smart gadgets are getting shrewdness aggressively during recent years [1]. Even a simple wrist watch is evolved as a health wrist band which is loaded with a couple of sensors onboard to track the wearer's activities and to monitor his health [2]. It is also programmed to trigger emergency protocols during the uncommon precarious health condition of the owner [3].

In earlier days, these kinds of devices were following multifarious hardware and communication protocols to get connect with other devices. The interconnection between those devices is complicate because of the amorphous communication architectures. This problem is solved by the IoT architecture in which most of the communications are carried out through IEEE 802.11 b/g/n standard. The uniformity provided by the IoT enables the devices to connect with each other seamlessly without much human- to-device interaction [4].

In recent times, these smart gadgets are being utilized to establish a smart home organization and the ensemble of the smart houses forms a smart city [5, 6]. A smart city is a well civilized association in which the natural resources are conserved prudently to serve the overall population of the city. Likewise, the environmental preservation is given higher priority to protect the city from undesirable pollutions caused by any industry, organization and cumulative transportation activities [7].

Multitudinous sensors and devices are interconnected under the IoT architecture of a smart city. Most of these sensor devices are designed in a way to stream the input data about its environment. The accumulation of the data will create massive databases which is very completed to handle with the conventional communication and computational infrastructures. Fortunately, emerging trends in IoT and Cloud computing are amalgamated together to support the big data processing comfortably [8]. This combination provides a polish way to collect, store, manage and to automate the flow of big data seamlessly which is the vital prerequisite for smart city organizations [9]. The visualization of the present environment of a smart city is also improved significantly as the consequence of the IoT-Cloud based wireless sensor networks.

The IoT devices are deployed in large-scale throughout the downtowns to constitute a smart city in many aspects from irrigation to industrial automations. The wide-spread placement of the IoT wireless sensor nodes are vulnerable to the hackers and intruders. Any intrusion in the IoT networks of smart cities can create a chaos easily and the organization may collapse rapidly as the aftermath of the security glitch. Since IoT permits heterogeneous node compatibility, it is a very crucial to design a dependable flawless network security scheme to prevent the hacking activities. Involving high security protocols in this province will affect the performance of the low powered IoT devices whereas low security protocols can be shattered easily [10]. Security Key generation, exchanging and invalidation are the substantial tasks in determining the security and power consumption of a security protocols. Therefore, scheming an impregnable counterbalanced lightweight security key handling mechanism is a persistent demand in IoT based Wireless Sensor Networks.

2. Existing methods

The There are some existing methods that servers the purpose of contributing for the lightweight security key handling in practice which are comparable to the proposed method. Hash-Based Conditional Privacy Preserving Authentication and Key Exchange Protocol Suitable for Industrial Internet of Things [11], Noisy Vibrational Pairing of IoT Devices [12], Blending Physically Unclonable functions with Identity Based Encrytption for Authentication and Key Exchange in IoT's [13], IoT-friendly AKE: forward secrecy and session resumption meet symmetric-key cryptography [14], A Lightweight Authentication and Key-Exchange Mechanism for 6LoWPAN-based Internet of Things [15], New Enhanced Authentication Protocol for Internet of Things [16], PUF Based Authenticated Key Exchange Protocol for IoT Without Verifiers and Explicit CRPs [17], Secure Mutual Authentication and Key-Exchange Protocol Between PUF-Embedded IoT Endpoints [18] and Sensing as a service in Internet of Things: Efficient authentication and key agreement scheme [19] are the existing methods chosen over here to compare with the proposed method in terms of standard network performance metrics.

2.1 Hash-based conditional privacy preserving authentication and key exchange protocol suitable for industrial internet of things (HBCPPA)

HBCPPA work is introduced to provide a robust hash-based conditional light weight protocol for privacy preservation with low computational overhead. The base security scheme of HBCPPA is derived from Elliptic Curve Cryptography (ECC). The key agreement process of HBCPPA uses hidden and dynamic modulus to improve the authentication security. Pseudonyms of different sensor nodes are used in the communication to enhance the anonymity if the sensor nodes involved in the communication including sender and receiver nodes. The nodes are restricted to use dynamic session modulus in which every node has to generate its own unique session specific modulus to converge with the common session key. The security analysis is performed by the combination of AVISPA and Real-or-Random Oracle Model. Security is analyzed against various well-known attacks such as Known session specific Temporary information attack, Computational DoS attack, Perfect forward secrecy attack and Replay attack. The Hash and X-OR operations used in HPCPPA supports the security improvements which is the advantage of this method, whereas, the dynamic session modulus and unique session specific modulus calculations affects the throughput of the network. The throughput values are comparatively lesser than the other methods which is known as the limitation of this work.

2.2 Noisy vibrational pairing of IoT devices (NVPID)

NVPID method provides security to the IoT sensor nodes by cloaking the vibration sounds which are generated by the IoT communication hardware during the pairing phase. Eavesdropper is a kind of device which can capture audio signal leakages or interference engendered during any radio frequency

communication. The vibration cloaking mechanism of NVPID protects the IoT nodes from the intruders by debasing their eavesdropper devices. NVPID is capable of using onboard speakers which are already exist in the IoT sensor devices. IoT sensor devices without speakers are to be added with nee speakers to apply NVPID. The main module of NVPID is the Vibration based pairing protocol which consists of audio leakage cancellation and audio leakage masking. NVPID is tested against proximity attack on vibration pairing. The implementation of NVPID requires only minimal computational overhead accession, which means there will be no significant reduction in performance metrics of the IoT network. Very low computational overhead and power consumption are the advantages of this method and vulnerability against several attacks other than proximity attack is known as the limitation.

2.3 Blending physically unclonable functions with identity based encryption for authentication and key exchange in IoTs (BPUFA)

Physically Unclonable Functions (PUF) is an essential hardware primordial to generate unique keys where large number of IoT devices participate in a network. The traditional authentication systems use a group of credentials with the password and digital certificates as the proof of authentication. Since IoT is a modern framework which can overcome the human-to- computer interdependence, conventional authentication protocols require significant amount of changes to be used. BPUFA uses Identity based Encryption (IBE), PUFs and Keyed Hash Function (KHF) in the authentication process during session establishment. ECC is used as the base of Public Mathematical Parameters in BPUFA to ensure bi-linearity, non-degeneracy and computability. BPUFA is implemented using Digilent Nexys-4 FPGA evaluation board which is power by Xilinx Artix-7 processor. Improved security is known as the advantage of this method and excess power consumption make it hard to apply for battery operated node, which is observed as the limitation of this method.

2.4 IoT-friendly AKE: forward secrecy and session resumption meet symmetric-key cryptography (IAKE)

IAKE portrays a third-party authenticated key exchange protocol to ameliorate forward security. The third-party authenticated key exchange protocol is designed based on symmetric key functions. In IAKE, session renewal is achieved without security compromising security in an IoT network. IAKE deals with authentication key server, application end device, communication server and the application server. Security management, Cryptographic separation, Server connection security and Quick session establishment are the important features offered by IAKE. Application security layer and Communication security layers are handled independently by this IAKE method to improve authentication security. The security model of IAKE is based on existing 2-Authenticated Key Exchange (2- AKE) model and Authenticated Confidential Channel Establishment (ACCE) [20]. Secured 3-party session establishment and session renewal are the advantages of IAKE method whereas elevated power consumption is known as the limitation of this method.

2.5 A lightweight authentication and key-exchange mechanism for 6LoWPAN-based internet of things (LAKEM)

Reducing the computational complexity of the conventional three factor authentication mechanism is the aim of LAKEM Method. Making use of the hash functions during the setup and registration phase is the primary technique used in LAKEM method to reduce the computational complexity. Since most of the IoT nodes are resources constrained, application of LAKEM reduced the computational cost and resource overhead. Provision for Manual authentication, Confidentiality and Integrity are the delivered features of LAKEM work. The security against Replay attack, Man-in-Middle Attack, Node Compromised Attack and Sybil Attacks are evaluated in this work. The claimed security validation is proved in LAKEM work by ProVerif Tools and with Burrows-Abadi-Needham logic. Sensible utilization of computational resources and higher security are the observed advantages of LAKEM method whereas, performance issues such as decay in overall throughput, increased communication delays and packet delivery rate are observed as the limitations.

2.6 New enhanced authentication protocol for internet of things (NEAP)

NEAP work concentrates on the New sensor addition phase, User Registration Phase, Login and Authentication Phase, and Password changing phase. In New sensor addition phase, the gateway generates a random and unique identification number and a dedicated key for every new sensor node. A database of these IDs and Keys are maintained in the gateway. In user registration phase, the new node sends its ID and Key along with two numbers

over a secured channel. Then a five-step authentication method is followed in the Login and Authentication phase to initialize the public channel communication between the gateway and the registered node. Password changing process is initialized by the user by logging in with the old password in public channel and by providing the authentication session key. NEAP claims that it is providing security against Password guessing attack, Insider Attack, Replay attack, Denning-Sacco attack, Stolen verifier attack, and Denial of Service attack. The complete NEAP work is described theoretically with sufficient equations. A transparent evaluation with any benchmark protocol analyzer or network simulator is not carried out which is known as the downside of NEAP work. The impact of the NEAP work in IoT networks performance is also not evaluated in this work.

2.7 PUF based authenticated key exchange protocol for IoT without verifiers and explicit CRPs (PAKEP)

Physical Unclonable Function (PUF) is a notorious technique getting popular these days which is used to generate unique identities for millions of different nodes in the network. PAKEP method uses PUF as a base to create the session keys between the nodes without involving server or verifier. An adversarial model in introduced in PAKEP model for secured IoT communication and validated using random oracle model. PAKEP consists of Initialization phase, Data provider registration phase, IoT node registration phase, and Authentication key exchange phase. A dedicated Secured Credential Generator (SCG) is also introduced in PAKEP work that plays a vital role in all the four phases. The evaluation of PAKEP model is performed based on theoretical heuristic proofs against Replay attack, Man-in-the-middle attack and forgery attack. Improved security is claimed as the advantages of PAKEP work, at the same time, absence of performance analysis of standard network parameters and utilization of proper network security measurement tool is understood constraints.

2.8 Secure mutual authentication and key-exchange protocol between PUF-Embedded IoT endpoints (SMAKE)

SMAKE work enables authentication and communication between two resource constraint nodes without storing Challenge-Response Pairs (CRP). Maintaining large number of CRPs is one of the fundamental tasks while using PUF technique. SMAKE addresses this issue by introduction a reverse fuzzy extractor to offload the resource intensive task to the server. SMAKE introduces committed functionalities to handle Enrollment phase and Mutual Authentication Key exchange phase. Device-side Resource and security are managed on SMAKE model using PUF, Reverse Fuzzy Extractor and Device Refresh modules. Reliability and Secrecy of SMKE work is evaluated using ProVerif software. Resource sensible improved security is the advantage of SMAKE work however higher jitter and latency are given as the limitations this work.

2.9 Sensing as a service in internet of things: efficient authentication and key agreement scheme (EAKAS)

EAKAS work enables the owners of the sensing devices to lend their devices as the platform Sensing as a Service. This comes under the Infrastructure as a service but with a short-term contract. EAKAS data session is established and protected using Fuzzy extractor, Elliptic Curve Diffie-Hellman algorithm, Symmetric encryption and Hash functions. Service Pre-deployment phase, User registration phase, Login Phase, User authentication and Key agreement phase, and Cloud service / Fog / Sensor authentication and key agreement phase are the different phases handled by EAKAS method. Automatic Validation of Internet Security Protocols and Applications (AVISPA) is used to evaluate the resistance of EAKAS work against Stolen user terminal attack, Eavesdropping attack, Replay attack, Man-in-the-middle attack, Impersonation attack, Node capture attack, Cloud attack, Denial-of-Service attack and Offline password guessing attack. High Level Protocol Specification Language (HLPSL) is used as the script language to perform AVISPA evaluation.

Improved security against a wide range of attacks is the stated advantage of EAKAS method whereas, the network performance such as throughput, communication delays and Packet delivery ratio are observed as the limitations.

An outline about the methodologies used in the existing methods, their merits and limitations are presented as a table given below.

3. Related works

There are two imperative concepts related to the proposed method's descriptions. They are Rosenberg Strong Pairing and Miller's algorithm on curves. Having a brief introduction to these concepts

Table 1. Summary of methodologies, Merits and limitations of existing methods

Author	Work	Methodology	Advantages	Limitations
Swapnil Paliwal [11]	Hash-Based Conditional Privacy Preserving Authentication and Key Exchange Protocol Suitable for Industrial Internet of Things	Robust Hash	Security	Low Throughput
S Abhishek Anand, Nitesh Saxen [12]	Noisy Vibrational Pairing of IoT Devices	Noisy Vibration Pairing	Security against noise tone attacks	N/A for all attacks
Urbi Chatterjee [13]	PUF+ IBE: Blending Physically Unclonable Functions with Identity Based Encryption for Authentication and Key Exchange in IoTs	Physically unclonable Functions	Security	High power consumption
Gildas Avoine [14]	IoT-friendly AKE: Forward Secrecy and Session Resumption Meet Symmetric-key Cryptography	Three party authentication	Security	Background key calculations drain nodes power faster
LekiChom Thungon [15]	A Lightweight Authentication and Key-Exchange Mechanism for 6LoWPAN-based Internet of Things	Hash based three factor authentication	Security	Performance decay
MouradeAzrour [16]	New Enhanced Authentication Protocol for Internet of Things	Five step authentication	Security	High communication delays
Yun-Hsin Chuang [17]	PUF Based Authenticated Key Exchange Protocol for IoT Without Verifiers and Explicit CRPs	PUF based Secured Credential Generator	Security against multiple attacks	Level of security
Yue Zheng [18]	Secure Mutual Authentication and Key-Exchange Protocol Between PUF-Embedded IoT Endpoints	Reverse Fuzzy Extraction based offloading	High security	High Jitter and Latency
Atef Bentahar [19]	Sensing as a service in Internet of Things: Efficient authentication and key agreement scheme	Symmetric encryption and hash functions	High security	Low Throughput and PDR

will facilitate the exposition of proposed method in an evident manner.

3.1 Rosenberg strong pairing

Pairing is a mathematical concept which refers the process of combining or encoding uniquely two natural numbers into a single natural number. The pairing function should have an unpairing function which refers the inverse pairing function. The invers pairing or unpairing function will retrieve corresponding source numbers from the paired number. This pairing concept can be applied in set theory to verify that integer number and rational number shave identical equality as natural numbers.

There are several pairing functions are available such as Cantor Pairing Function, Rosenberg Paring Function and Elegant Pairing function. Rosenberg-Strong Pairing function is opted in this work due to its advantages over the other methods in higher dimensions [21]. The Rosenberg Strong Pairing function is given in Eq. (1).

$$z = RosenbergPair(x,y) = (max(x,y)) + max(x,y) + x - y \quad (1)$$

where x, y are the input numbers to be encoded and z is the encoded result.

The Rosenberg Strong Pairing function number mapping is illustrated in the following Figure.

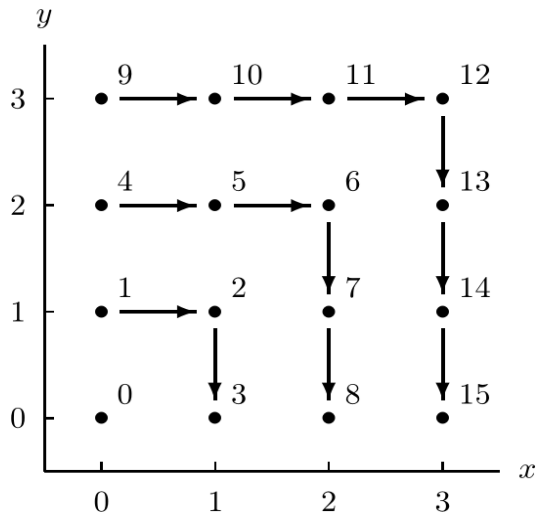

Figure. 1 Rosenberg strong pairing number map

The Rosenberg Strong Unpairing function is given in Eq. (2).

$$(x,y) = RosenbergUnpair(z) = RosenbergPair^{-1}(z) = \begin{cases} (z-m^2, m)\ if\ z-m^2 < m \\ (m, m^2+2m-z)\ \ Otherwise \end{cases} \quad (2)$$

where m is calculated as $\lfloor\sqrt{z}\rfloor$.

The basic functionality of Rosenberg Strong Pairing function is pursued in some segments of the proposed method.

3.2 Miller's algorithm on curves

Miller's algorithm confides on evaluating functions of elliptic curves. A comprehension about functions and divisors of elliptic curve is crucial to describe Miller's algorithms. The functions and divisors of a line is given below as a primer for the same in elliptic curves.

Let $\overline{K}$ be an algebraically closed field with an affine line A^1. Adding the point at infinity brings the projective line $\mathbb{P}^1 = A^1 \cup \{\infty\}$. Then the rational functions $\overline{K}$ ($\mathbb{P}^1$) on A^1 can be represented as $\overline{K}(t)$. The prime number rational function f is provided as follows.

$$f = \frac{P}{Q} = \frac{M(t-x_i)^{n_i}}{M(t-y_i)^{m_i}} \in \overline{K}(t) \quad (3)$$

where P, Q are relatively prime numbers, x_i are 0s of f with multiplicity m_i, y_i are the poles of f with multiplicity ni

Based on Eq. (3), multiplicity $ord(f)$ for each point $x \in \mathbb{P}^1(\overline{K})$ can be defined as follows

$$ord_x(f) = n\ if\ x\ is\ a\ zero\ of\ f\ with\ multiplicity\ n - n\ if\ x\ is\ a\ pole\ of\ f\ with\ multiplicity\ n0\ \ Otherwise \quad (4)$$

The change of variables $u = \frac{1}{t}$ sets $\infty \to 0$ to calculate the evaluation of f at ∞. Let g be the genus by $g(u) = f^{-1}(u)$ to relate $f(t) = g(u)$ when $t = \frac{1}{u}$. By this definition, the value of f on ∞ comes to g on 0. Then the formal sum to the function f is as follows

$$div(f) = \sum_{x \in P^{-1}(K)} ord_x(f)[x] \quad (5)$$

where $[x]$ refers the point $x \in \mathbb{P}^1(\overline{K})$ in the formal sum which is the divisor of f.

If f_1 and f_2 are the rational functions in a way that $di(f1) = div(f2)$, then they will have same zeros and poles with different multiplicative constant. By this statement, a divisor D can be defined as a formal sum of finite number of points as follows

$$D = \sum_{x \in P^{-1}(K)} n_i[x_i] \quad (6)$$

By Eqs. (3) to (6), the Miller's algorithm is constituted as follows. Let E be an elliptic curve with the principle divisor F as $f = div(f)$, then f is a unique constant and OE is either a pole or zero of f which depicts that $f(O_E) = 1$. Given that has the identical order at O_E as f, including the function is defined at O_E makes it possible to normalize f uniquely where the value is 1 at O_E. While defining f_F as the unique function such that $F = div(f_F)$ and $= 1$. As per this statement, if F is rational, then f_F should also be rational. The finalized Miller's pairing algorithm on Elliptic curve is given below.

$$\mu_{PQ} = \frac{y - a(x-xp) - yp}{x + (xP + xQ) - a^2} \quad (7)$$

Where the value of T is resolved using Eq. (8) given below

$$T = \begin{cases} \frac{H'(xP)}{2yP}\ \ if\ P = Q \\ \frac{yP - yQ}{xP - xQ}\ \ Otherwise \end{cases} \quad (8)$$

Where H' is the inverse height function of the elliptic curve.

4. Proposed method: masked location based key exchange mechanism (MLKEM)

The proposed method contains two essential components, they are Dual Rosenberg Pairing Location Masker (DRPLM) and Fuzzy Miller's Elliptic Curve Key Exchange (FMECKE). MLKEM is the integration of these methods in which DRPLM is used to initialize the communication session keys in a unique astute way and FMECKE is used to distributing the keys in a protected manner.

4.1 Dual rosenberg pairing location masker (DRPLM)

DRPLM is used to mask the location and the MAC address of the IoT Node into a single number which will be used as the authentication key to initialize a communication session. The location of IoT node is delineated using the latitude - longitude

representation and the hardware MAC address is unique for all network nodes. The latitude and longitudes are noted as a geographical point (00.0000, 00.0000) which consists two 8 digits real numbers with 6 precision digits. The MAC addresses are represented in 6 segment 2-digit hexa-decimal numbers in the range from 00H to FFH.

Since a network with some IoT nodes can have some battery operated low computational powered devices, handling floating point numbers and large integer values is a misery for them. Two consecrated hash functions are introduced in this proposed method to regulate the number of digits of geographical location (Geo-Hash HG) and MAC address (MAC-Hash HM) into 2-digit integer numbers. The digits reduction hash functions are designed with 'high bit- processing & low-mathematical calculation' mode which is also known as combined diffusion to reduce the computation complexity during the session key calculations.

4.1.1. Geo-Hash

Geo-Hash function is used to combine the 12-digit latitude and longitude values into a two-digit values using bit- swapping and combining procedure. Each input digit is converted as packed Binary Coded Decimal (BCD) and divided into 3 packed BCD blocks for both latitude and longitude individually. Let the digits of latitude be (α5α4.α3α2α1α0), then the packed BCD blocks will be noted as{α5α4}{α3α2}{α1α0}. A single bitwise diffusion of Geo- Hash is Given in Fig. 2
The result of this diffusion process is represented as {α_5, α_4}{α_3, α_2}{α_1, α_0}. Similarly, for longitude the source digits {β_5, β_4} {β_3, β_2}{β_1, β_0} will be diffused as{β_5, β_4}{β_3, β_2}{β_1, β_0}. Both the bitwise diffused latitude and longitude blocks are further fed to the block wise diffusion process. The block-wise diffusion process takes care of shuffling the data blocks between latitude and longitude. The architecture of block-wise diffusion process is given in Fig. 3.

Let the result of block-wise diffusion be {γ5, γ4, γ3, γ2, γ1,0}, the combination process computes the final result HG as follows.

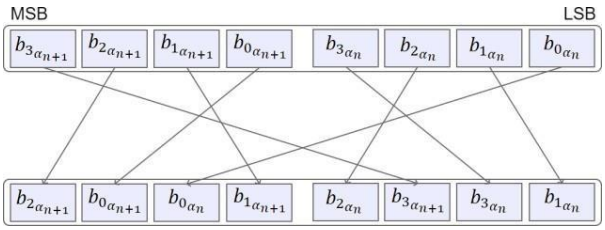

Figure. 2 Bitwise diffusion

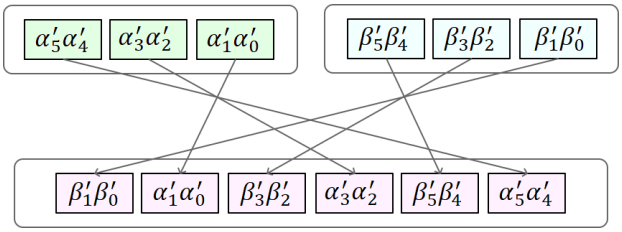
Figure. 3 Block-wisediffusion

$$HG=\gg((y0\&y1) + (y2 \oplus y3)) |(y4\&y5) \quad (9)$$

Where the symbol $\gg$ refers the bitwise right-shift operator.

4.1.2. MAC-Hash

The MAC-Hash function includes a new block-wise diffusion and the bit size reduction processes. The MAC addresses are represented in 12-digit hexadecimal numbers split into 6 equal 2-digit segments as 00:00:00:00:00:00. The first three segments indicate the Unique Manufacturer Identifier and the remaining segments denotes the network interface controller specific identification numbers. The MAC address is unique for every network device that connects with the internet through IEEE 802.11 ac/b/g/n standards. Let {δ5, δ4, δ3, δ2, δ1, δ0} be the equational representation of the MAC address, then the block wise diffusion is performed as in the following figure.

The result $\{s_5, s_4, s_3, s_2, s_1, s_0\}$ from the block-wise diffusion is combined to achieve MAC-Hash function H_M value as by the following equation.

$$H_M = \left(\sum_{i=0,2,4}(\delta_i' \oplus \delta_{i+1}')\right) \gg 3 \quad (10)$$

Where the symbol $\oplus$ refers the X-OR operation and the symbol $\gg$ refers the bitwise right-shift operation.

Then the location masked MAC address authentication key ka is generated by the DRPLM through the following equation.

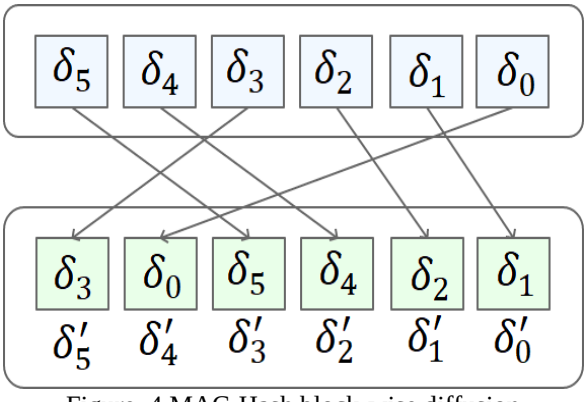
Figure. 4 MAC-Hash block-wise diffusion

$$ka = (max(HG, HM)) + max(HG, HM) + HG - HM \quad (11)$$

By this way, the session authentication key is generated by the IoT network nodes associated in the wireless environment.

4.2 Fuzzy Miller's elliptic curve key exchange (FMECKE)

FMECKE is segregated into two sub-modules as Fuzzy Computational Complexity Determiner (FCCD) and Miller's Elliptic Curve Key Exchange Procedure (MECKEP). FCCD is used to find out the optimum key sizes to be involved in the encryption and MECKEA is used to exchange the key between IoT nodes and the gateway.

4.2.1. Fuzzy computational complexity determiner (FCCD)

The computational countenance in terms of processing speed, committed memory for key calculations and the power status of the nodes are given as the input to the FCCD to get the optimized key size as the output [22]. The first step of FCCD is calculating the computational power quotient Φ using the following equation.

$$\Phi = 2^n \frac{\sigma_m}{\sigma_{m_{max}}} + 2^{n+1} \frac{p}{\sigma_{p_{max}}} \quad (12)$$

Where n is the resource priority index starts with 0, σ_m is the available memory of the device, σ_p is the processing power of the device, $\sigma_{m_{max}}$ is the maximum memory availability of a node in the entire network and $\sigma_{p_{max}}$ is the maximum processing power of a node in the entire network. Communication scenarios between all low powered nodes (LOW), either low powered node (MEDIUM) and all high-powered nodes (HIGH) are covered in the FCCD operation. The communication category $Ccat$ is classified using the following fuzzy rule.

$$C_{cat} = \begin{cases} HIGH\ if\ \Phi \geq \frac{1}{2}\left(\sigma_{p_{max}} + \sigma_{m_{max}}\right) \\ MEDIUM\ if\ \Phi \geq \frac{1}{2}\left(\sigma_{p_{max}} + \sigma_{m_{max}}\right) \\ \quad and \\ < \frac{1}{2}\left(\sigma_{p_{max}} + \sigma_{m_{max}}\right) \\ LOW\ Otherwise \end{cases} \quad (13)$$

The size of the key is permitted to be 256-bits for HIGH category, 128-bits for MEDIUM category and 64-bits for LOW Category for the consecutive functional modules.

4.3 Miller's elliptic curve key exchange procedure (MECKEP)

A distinct version elliptic curve point generation procedure is used in this module [23]. The point generation relies on three numbers namely p is a prime number, a and b are two random integers. The prime number p is selected randomly based on the value of $ccat$. The values of a and b are randomly selected in a way to satisfy the condition $4a^3 + 27b^2 \neq 0$ for a $y^2 = x^3 + ax + b$ format elliptic curve. The elliptic curve points are generated into several tuples represented as (x, y) using the following equations.

$$\kappa = (x^3 + ax + b) mod\ p \quad (14)$$

$$\lambda = \left(\kappa^{\frac{p-1}{2}}\right) mod\ p \quad (15)$$

The value of y is calculated for given x as by the following equation

$$y^2\ mod\ p = \kappa \quad (16)$$

The values of HG and HM are truncated to 0xFF through a single and triple right shift bitwise-operations in order, a group of 256 elliptic curve points are generated to proceed with the key exchange process. The values of the x tuples in the generated points are substituted from the value 0x00 to 0xFF which is treated as the replica of the session key ka to get the substitute key ks from the y of the same tuple.

The value of ks is sent to the MAG (Mobile Access Gateway). MAG computes Geo-Hash and the MAC-Hash values of the sender since the values of p, a, b of ECC and Dual-Rosenberg Location Masking Equation (Eq. (11)) are pre-determined in the MAG. If the session request of the sender is valid, then MAG pings the corresponding received. Receiver follows the same steps to acknowledge the mag with its session key. MAG replies sender by supplying the receivers key using MECKEP and senders key to the receiver.

The communication flow is given below in simple steps

Step 1: Let k_s^S and K_s^R be the session keys of sender and receiver respectively

Step 2: Sender sends the value of k_s^S to the MAG

Step 3: MAG calculates the Geo-Hash and MAC hash for the sender (Since reversing the HASH is costly) Step 4: MAG validates the sender's details through k_s^S

Step 5: In case of malicious session request, MAG generates a security menace and discards the session request
Step 6: If the request is valid, MAG pings the receiver
Step 7: Receiver sends the value of k_s^R to the MAG
Step 8: MAG calculates the Geo-Hash and MAC hash for the receiver Step 9: MAG validates the sender's details through k_s^R
Step 10: In case of malicious session request, MAG generates a security menace and discards the session request
Step 11: MAG supplies k_s^R to the sender and k_s^S to the receiver using MECKEP and the session initialization will be a success

5. Experimental setup

OPNET – one of the best network simulation and evaluation tools of the decade which is developed by OPNET Technologies Inc. and acquired by Riverbed Technology. OPNET uses graphical representations of different network nodes and network environments [24, 25]. It has the provision to inherit the real-world network environments by defining the latitude and longitude details. OPNET permits to define and override the default network node types, protocols and network communication strategies. OPNET has an advanced property of processing C++ codes to define the network strategies such as in Automatic Validation of Internet Security Protocols and Applications (AVISPA) [26] with High Level Protocol Specification Language (HLPSL).

Experiments are carried out in OPNET repeatedly with different number of nodes for existing and proposed methods. The Simulation world details are provided in Table 2. Visual Studio is one of the Industry leading Integrated Development Environments (IDE) from Microsoft. Visual Studio [27] is used to code the Network scripts and a dedicated UI is designed to perform repeated OPNET simulations, acquire results and to plot the comparison graphs. User Interface screen image is given in Fig. 5. Network node placement in OPNET Simulator is given in Fig. 6.

Table 2. Simulation parameters

S.No	Entity	Details
1	Simulation Area	10000 Square meters
2	Number of Nodes	100 to 1000 in step 100
3	IoT-Node types	ESP-32, ESP-8266, LoRa (Uniform Distribution)
4	Number of Routers	Automatic Selection
5	Node Placement	Random distribution
6	Network density	Default
7	RF Range of IoT-WSN Nodes	Based on the type from 100 meters to 1000 meters
8	Frequency bands	Auto-select
9	Simulation Time	168 real-world hours

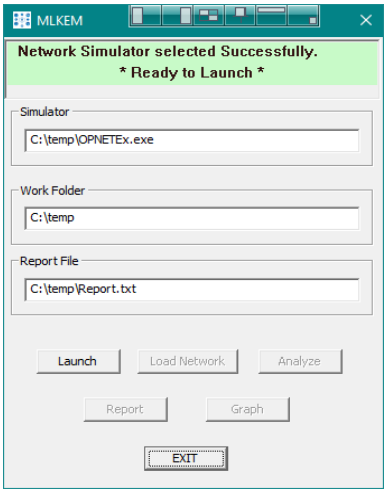

Figure. 5 Dedicated user interface

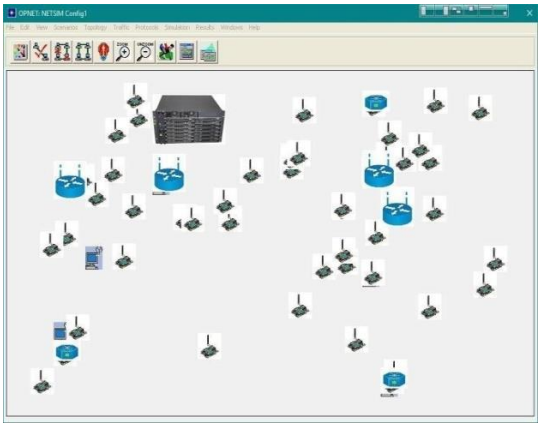
Figure. 6 Node placements OPNET

6. Results and analysis

Regular network performance assessment metrics such as Throughput, Communication Delays, Packet Delivery Ratio, Security and Energy consumption are measured through the OPNET Simulation. Observed results are discussed below by tabulating the values and by plotting as graphs.

6.1 Throughput

Throughput refers the successful data communication in a network channel. Throughput is measured in bits-per- second (bps) units which represents how fast the communication occurs in the channel. In general, IoT wireless sensor nodes are communicating little pieces of information over the network whereas, broadened use of IoT devices such as healthcare monitors continuously streaming data to the network. The tremendous increase of

Table 3. Throughput

Throughput (kbps)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	27439	28575	32219	35509	35730
200	26073	27733	31093	34315	34543
300	24737	26211	30380	33053	33298
400	23713	25451	29235	31962	32490
500	22795	23922	27948	30962	30914
600	21485	23087	26864	29970	30034
700	20032	21740	25861	28879	29071
800	19096	20565	24607	27552	27627
900	17679	19453	23829	26335	26568
1000	16719	18566	22688	25550	25550
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	36457	36698	36384	36840	37156
200	34927	35460	34844	35698	36487
300	33933	34512	33972	34584	35205
400	32926	33631	32914	33477	34162
500	31633	32436	31678	32452	33164
600	30222	31041	30624	30981	32322
700	29385	30072	29455	30126	31459
800	27877	29297	28549	28495	30515
900	26697	27914	27608	27476	29275
1000	25661	26797	26180	26162	28120

Table 4. Latency

Parameter: Latency (mS)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	28	24	22	20	30
200	28	27	26	24	30
300	30	29	25	22	30
400	33	28	27	23	30
500	34	31	27	24	32
600	32	31	27	25	33
700	33	31	30	28	33
800	37	32	30	27	36
900	35	34	30	28	36
1000	36	36	31	32	37
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	30	23	35	46	15
200	33	26	37	46	16
300	34	26	38	47	18
400	35	26	39	46	21
500	34	26	40	47	22
600	37	28	41	50	22
700	38	29	43	49	22
800	38	29	41	52	25
900	40	30	45	53	25
1000	41	32	46	54	24

individual healthcare devices causes plenty of data flow over the network, so it is important to measure the throughput and to find the maximum data transfer capacity of the network. Throughput values are measured for 100 to 1000 number of nodes with different methods are given in Table 3.

In accordance with observed results, it is realized that the Throughput values of MLKEM is higher than other methods for different node count network scenario. The highest throughput 37156 Kbps is achieved by MLKEM during the experiment with 100 number of nodes. The throughput average of MLKEM is 32786 Kbps followed by EAKAS and PAKEP with the throughput average values of 31629 Kbps and3178 ordered based on the performance. The comparison graph of throughput results is given below in Fig. 7.

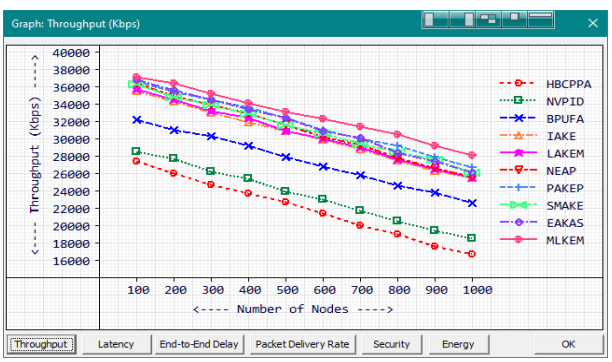

Figure. 7 Throughput

6.2 Latency

Latency is the duration between a data transfer request and the beginning of the data transfer. If the latency is higher, then the overall response time of the network will be high. Therefore, a good communication protocol should take fewer latency values. Latency values are measured for existing and proposed methods and given in Table 4.

The observations point that the latency values of the proposed method are lower than the other methods in comparison. The latency averages of MLKEM, IAKE, BPUFA, PAKEP, NVPID, HBCPPA, LAKEM, NEAP, SMAKE and EAKAS are 21 mS, 25.3 mS, 27.5 mS, 27.5 mS, 30.3 mS, 32.6 mS, 32.7 mS, 36 mS, 40.5 mS and 49mS. The first three low latency methods are MLKEM, IAKE

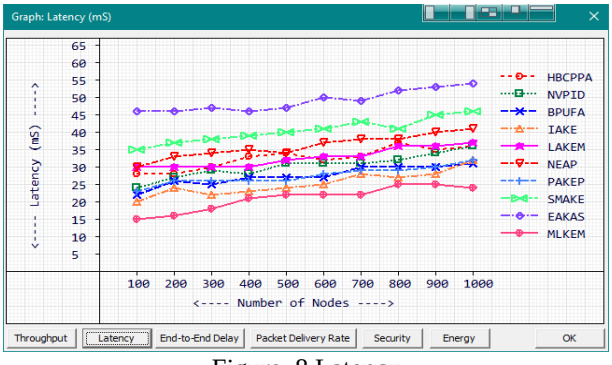

Figure. 8 Latency

Table 5. End-to-End delay

Parameter: End-to-End Delay (mS)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	115	109	98	101	110
200	119	112	110	102	115
300	124	108	113	108	121
400	130	117	112	108	118
500	132	121	118	108	123
600	130	119	116	111	130
700	137	121	120	118	131
800	145	127	126	125	130
900	145	131	129	123	138
1000	143	130	131	127	142
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	112	105	121	145	93
200	115	106	131	151	99
300	118	115	135	147	105
400	128	116	136	152	109
500	125	115	133	152	111
600	133	116	143	161	113
700	138	120	147	166	117
800	138	122	147	169	113
900	145	132	144	167	116
1000	140	135	147	173	118

and BPUFA with the minimum latency readings of 15mS, 20mS and 22mS during the execution with 100 number of nodes. The highest latency of MLKEM method is only 25mS which is measured during the execution with 700 and 800 number of nodes. Based on the observed results, the highest latency of MLKEM is comparable lower than all the other methods involved in the evaluation process. The latency comparison graph is given in Fig. 8.

6.3 End-to-End delay

End-to-End delay is the time duration between the beginning of a data packet transfer from the source node and ending in the destination node. It consists of all communication delays such as latency, IP delay, system delay and jitter. End-to-End delay also should be kept in control to design a better network protocol. Measured values of End- to-End delays for different methods are given in following Table 5.

The End-to-End delay averages of MLKEM, IAKE, BPUFA, PAKEP, NVPID, LAKEM, HBCPPA, SMAKE and EAKAS are 109.4 mS, 113.1 mS, 117.3 mS, 118.2 mS, 119.5 mS, 125.8 mS, 129.2 mS, 132 mS, 138.4 mS and 158.3 mS given in order.

The lowest End-to-End delay is achieved by MLKEM method which is 93 mS recorded during

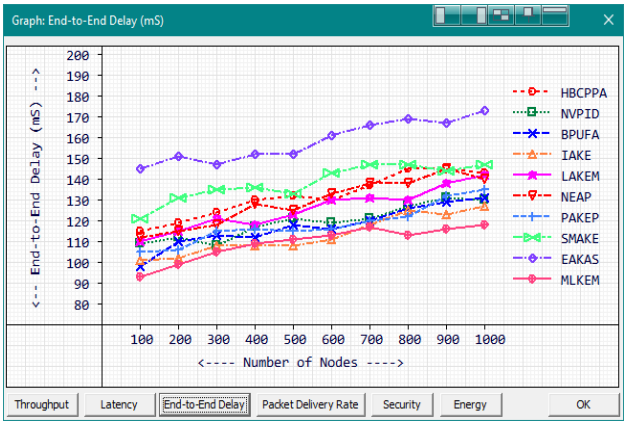

Figure. 9 End-to-End delay

the simulation with 100 number of nodes. The highest end-to-end delay value of MLKEM is 118 mS observed during 1000 number of nodes. Just a negligible increase of 25 mS happened in MLKEM while raising 100 to 1000 nodes in the network environment. This least quantity augmentation shows the ability of MLKEM while applying in scalable networks. The comparison graph of End-to-End delay is given in following Figure.

6.4 Packet delivery ratio

Packet Delivery Ratio (PDR) is the ratio between number of transmitted data packets from the source node and number of successfully received data packets by the destination node. Higher value of PDR refers low number data collisions and packet drops. Packet Delivery Ratio values for methods HBCPPA, NVPID, BPUFA, IAKE and MLKEM are given in Table 6 and the comparison graph is given in Fig. 10.

Proposed MLEKM method achieved the highest PDR averages during the experiments. The PDR averages of MLKEM, PAKEP, LAKEM, SMAKE, NEAP, EAKAS, IAKE, BPUFA, NVPID and HBCPPA are 95.2%, 94.1%, 93.9%, 92.9%, 92.9%, 92.1%, 91.1%, 90.2%, 90.1% and 87.1 respectively. MLKEM managed to score the highest PDR of 99% during the evaluation with 100 and 200 number of nodes in the network environment. MLKEM achieved 91% PDR with high density network environment with 1000 number of nodes – which is comparably higher than other methods.

Packet Delivery Ratio comparison graphs is provided below as Fig. 10.

6.5 Security

Security is one of the important metrics in network communication. The entire network can be in vulnerable situation if security is compromised.

Table 6. Packet delivery ratio

Parameter: Packet Delivery Ratio (%)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	91	94	95	96	98
200	91	94	94	95	97
300	90	93	93	94	96
400	89	91	91	92	96
500	87	91	90	91	94
600	87	90	90	91	93
700	86	89	89	89	92
800	84	88	88	88	92
900	84	86	87	88	91
1000	82	85	85	87	90
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	97	98	97	97	99
200	96	97	97	96	99
300	96	97	95	94	98
400	94	95	94	94	97
500	94	95	93	92	95
600	92	94	92	91	95
700	92	93	92	91	93
800	90	92	91	90	93
900	90	91	90	88	92
1000	88	89	88	88	91

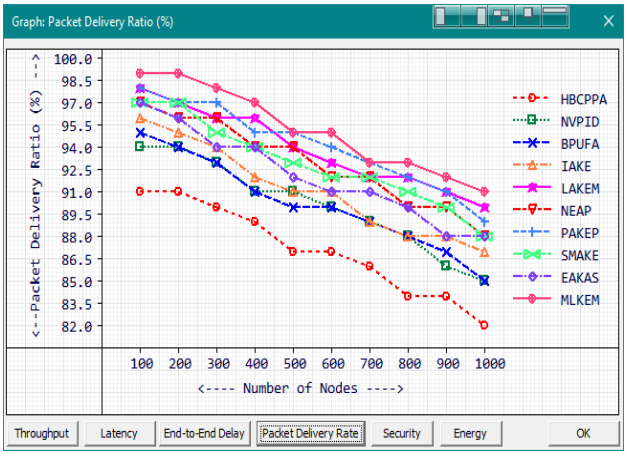

Figure. 10 Packet delivery ratio

IoT-WSN are used in several sensitive area such as healthcare and industrial automations, security is the prime aspect taken in to consideration. OPNET has the facility to measure the security in a network architecture by introducing random intruder attacks using the formula $100 - \bar{\rho} \times 100$ where ρ is the number $\rho + \bar{\rho}$ of rigid data packets and $\bar{\rho}$ is the number of compromised data packets. The measured security values of existing and proposed methods are given in Table 7.

MLKEM scored highest security score average of 98.6%. EAKAS, NEAP and LAKEM Methods got the equal security average score 97.5%. SMAKE, IAKE, PAKEP, BPUFA, HBCPPA and NVPID methods scored 96.4%, 96.3%, 95.6%, 95.3%, 94.6% and 92.7% respectively.

The highest security level 99% is achieved by MLKEM during the evaluation with 100, 300, 400, 500, 600 and 1000 number of nodes. EAKAS, NEAP and LAKEM methods are achieved 98% as the highest security scores.

The security comparison graph for the existing and proposed method is submitted in Fig. 11.

Table 7. Security

Parameter: Security (%)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	94	92	95	96	98
200	94	93	96	96	97
300	95	93	96	97	98
400	94	93	95	96	98
500	95	93	95	96	97
600	95	92	96	97	98
700	95	93	95	96	97
800	94	92	95	97	98
900	95	93	95	96	97
1000	95	93	95	96	97
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	97	95	97	98	99
200	97	96	96	97	98
300	97	95	96	98	99
400	98	96	97	97	99
500	97	95	96	97	99
600	98	96	97	98	99
700	98	96	97	98	98
800	98	96	96	97	98
900	98	95	96	97	98
1000	97	96	96	98	99

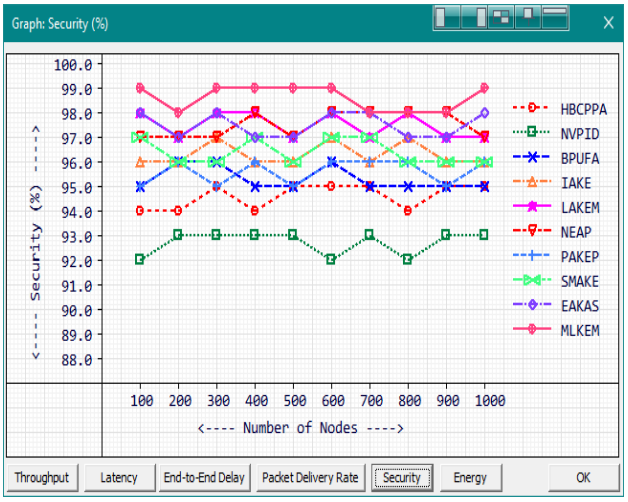

Figure. 11 Security

6.6 Energy

Energy consumption is one of the prime factors where there are a notable number of battery-powered network nodes. As the IoT based wireless sensor network nodes are primarily battery-operated devices, Energy consumption is one of the vital parameters here for the battery-operated devices. Energy is measured in Micro-Joules (uJ) and during the simulation, it is observed that the energy consumption to complete a secured network transaction increases along with the number of nodes.

The performance rank of compared methods based on energy efficiency is MLKEM, SMAKE, PAKEP, EAKAS, NEAP, LAKEM, NVPID, HBCPPA, IAKE and BPUFA with the energy consumption average values of 348.2uJ, 349.7uJ, 367.2uJ, 381 uJ, 388.7uJ, 396.1uJ, 588.9uJ, 640.1uJ, 698uJ and 828.3 uJ. The minimum energy consumption 292uJ is observed for MLKEM during the performance evaluation with 100 number of nodes.

The highest energy consumption of MLKEM is 407uJ is recorded during the simulation with 1000 number of nodes. The 115uJ increase in energy for the increase in 900 number of nodes is very nominal while comparing with other methods. The highest power consumption 894uJ is observed while simulating BPUFA method with 1000 number of nodes.

Table 8. Energy

Parameter: Energy (uJ)					
Nodes	HBCPPA	NVPID	BPUFA	IAKE	LAKEM
100	577	528	760	644	332
200	599	534	774	659	347
300	613	554	803	666	366
400	616	565	810	672	374
500	626	576	823	689	390
600	647	600	830	700	405
700	667	617	852	722	412
800	672	632	865	735	427
900	679	628	872	740	445
1000	705	655	894	753	463
Nodes	NEAP	PAKEP	SMAKE	EAKAS	MLKEM
100	335	301	280	321	292
200	334	324	310	332	306
300	362	332	308	348	309
400	366	355	333	368	318
500	383	360	346	370	349
600	402	364	354	386	355
700	400	380	367	395	371
800	419	406	388	414	381
900	440	419	392	435	394
1000	446	431	419	441	407

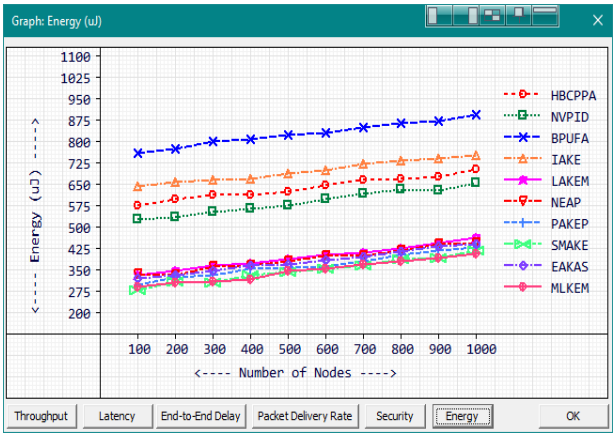

Figure. 12 Energy

The energy consumption values are given in Table 8.

Energy consumption result values are plotted as a graph– given as Fig. 12.

7. Conclusions

Improving network performance while sustaining security in confined resource heterogeneous network architecture similar to IoT is a challenging process. Preamble of Dual Rosenberg pairing location masker and Fuzzy Miller's elliptic curve key exchange modules are aggregated as MLKEM to achieve higher throughput and packet delivery ratio, which is the commenced novelty of this work. It is also observed during the experiments, that the communication delays are further reduced in proposed method without compromising security. The energy efficiency is obtained in MLKEM which is an added advantage for battery operated IoT devices. Based on the observation results, proposed MLKEM can be undertaken for ongoing and upcoming applications in various fields such as Agriculture, Smart city management, Environmental monitoring and real-time clinical database management systems.

Conflict of Interest

There is no conflict of interest between the authors.

Author Contributions

The contributions of authors are as follows:

A. Anandhavalli, Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation and Writing-original paper draft.

Dr. A. Bhuvaneswari : Validation, Supervision and Project Administration.

References

[1] L. Pawar, R. Bajaj, J. Singh, and V. Yadav, "Smart City IoT: Smart Architectural Solution for Networking, Congestion and Heterogeneity", In: *Proc. of International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 124-129, 2019, doi: 10.1109/ICCS45141.2019.9065688.

[2] A. Ukil, S. Bandyoapdhyay, C. Puri, and A. Pal, "IoT Healthcare Analytics: The Importance of Anomaly Detection", In: *Proc. of IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pp. 994-997, 2016, doi: 10.1109/AINA.2016.158.

[3] S. Pinto, J. Cabral, and T. Gomes, "We-care: An IoT-based health care system for elderly people", In: *Proc. of IEEE International Conference on Industrial Technology (ICIT)*, pp. 1378-1383, 2017, doi: 10.1109/ICIT.2017.7915565.

[4] S. A. Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review", In: *Proc. of International Conference on Information Technology (ICIT)*, pp. 685-690, 2017, doi: 10.1109/ICITECH.2017.8079928.

[5] Y. Kim, Y. Park, and J. Choi, "A study on the adoption of IoT smart home service: using Value- based Adoption Model", *Total Quality Management & Business Excellence*, Vol. 28, Issue 9-10, pp. 1149-1165, 2017, doi: 10.1080/14783363.2017.1310708.

[6] J. K. D. Barriga, C. D. G. Romero, J. I. R. Molano, "Proposal of a Standard Architecture of IoT for Smart Cities", *Uden L., Liberona D., Feldmann B. (eds) Learning Technology for Education in Cloud – The Changing Face of Education. LTEC. Communications in Computer and Information Science*, Vol. 620, pp. 77-89, 2016, doi: 10.1007/978-3-319-42147-67.

[7] J. I. H. Vega, E. R. Varela, N. H. Romero, C. H. Santos, J. L. S. Cuevas, and D. G. P. Gorham, "Internet of Things (IoT) for Monitoring Air Pollutants with an Unmanned Aerial Vehicle (UAV) in a Smart City", *Smart Technology, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, Vol. 213, pp. 108-120, 2018, doi: 10.1007/978-3-319-73323-411.

[8] X. Xu, Q. Liu, Y. Luo, K. Peng, X. Zhang, S. Meng, and L. Qi, "A computation offloading method over big data for IoT-enabled cloud-edge computing", *Future Generation Computer Systems*, Vol. 95, pp. 522-533, 2019, doi: 10.1016/j.future.2018.12.055.

[9] A. Yassine, S. Singh, M. S. Hossain, and G. Muhammadd, "IoT big data analytics for smart homes with fog and cloud computing", *Future Generation Computer Systems*, Vol. 91, pp. 563-573, 2019, doi: 10.1016/j.future.2018.08.040.

[10] H. R. Ghorbani and M. H. Ahmadzadegan, "Security challenges in internet of things: survey", In: *Proc. of IEEE Conference on Wireless Sensors (ICWISE)*, pp. 1-6, 2017, doi: 10.1109/ICWISE.2017.8267153.

[11] S. Paliwal, "Hash-Based Conditional Privacy Preserving Authentication and Key Exchange Protocol Suitable for Industrial Internet of Things", *IEEE Access*, Vol. 7, pp. 136073-136093, 2019, doi: 10.1109/ACCESS.2019.2941701.

[12] A. Anand and N. Saxena, "Noisy Vibrational Pairing of IoT Devices", *IEEE Transactions on Dependable and Secure Computing*, Vol. 16, No. 3, pp. 530-545, 2019, doi: 10.1109/TDSC.2018.2873372.

[13] U. Chatterjee and V. Govidan, "Blending Physically Unclonable functions with Identity Based Encryption for Authentication and Key Exchange in IoTs", *IEEE Transactions on Dependable and Secure Computing*, Vol. 16, No. 3, pp. 424-437, 2019, doi: 10.1109/TDSC.2018.2832201.

[14] G. Avoine, S. Canard, and L. Ferreira, "IoT-Friendly AKE:Forward Secrecy and Session Resumption Meet Symmetric-Key Cryptography", *Sako K., Schneider S., Ryan P. (eds) Computer Security – ESORICS 2019, Springer, Lecture Notes in Computer Science*, Vol. 11736, 2019, doi: 10.1007/978-3-030-29962-022.

[15] L. C. Thungon, N. Ahmed, S. C. Sahana, and M. I. Hussain, "A lightweight authentication and key exchange mechanism for IPv6 over low-power wireless personal area networks-based Internet of things", *Wiley Online Library*, Vol.32, Special Issue: Securing the Internet-of-Things: Advances, challenges, future trends, 2021, doi: 10.1002/ett.4033.

[16] M. Azrour, J. Mabrouki, A. Guezzaz, and Y. Farhaoui, "New enhanced authentication protocol for Internet of Things", *Big Data Mining and Analytics*, Vol. 4, No. 1, pp. 1-9, 2021, doi: 10.26599/BDMA.2020.9020010.

[17] Y. H. Chuang and C. L. Lei, "PUF Based Authenticated Key Exchange Protocol for IoT Without Verifiers and Explicit CRPs", *IEEE Access*, Vol. 9, pp. 112733-112743, 2021, doi: 10.1109/ACCESS.2021.3103889.

[18] Y. Zheng and C. H. Chang, "Secure Mutual Authentication and Key-Exchange Protocol between PUF-Embedded IoT Endpoints", *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5, 2021, doi: 10.1109/ISCAS51556.2021.9401135.

[19] A. Bentahar, A. Meraoumia, L. Bradji, and H. Bendjenna, "Sensing as a service in Internet of Things: Efficient authentication and key agreement scheme", *Journal of King Saud University - Computer and Information Sciences*, Vol. 34, Issue. 8, Part A, pp. 5493-5509, 2022, doi: 10.1016/j.jksuci.2021.06.007.

[20] K. Bhargavan, I. Boureanu, P. Fouque, C. Onete, and B. Richard, "Content delivery over TLS: a cryptographic analysis of keyless SSL", *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 1-16, 2017, doi: 10.1109/EuroSP.2017.52.

[21] M. P. Szudzik, "The Rosenberg-Strong Pairing Function", *Computer Science > Discrete Mathematics, ARXIV*, pp. 1-27, 2019, doi: 10.48550/arXiv.1706.04129.

[22] J. Ye, "Advances in Fuzzy Decision Theory and Applications", *Cognitive Science – Decision Theory, Research Gate*, 2021, doi: https://www.researchgate.net/publication/355474550_Advances_in_Fuzzy_Decision_Theory_and_Applications.

[23] K. H. Moussa, A. H. E. Sakka, S. Shaaban, and H. N. Kheirallah, "Group Security Authentication and Key Agreement Protocol Built by Elliptic Curve Diffie Hellman Key Exchange for LTE Military Grade Communication", *IEEE Access*, Vol. 10, pp. 80352-80364, 2022, doi: 10.1109/ACCESS.2022.3195304.

[24] Z. Lu and H. Yang, "Unlocking the Power of OPNET Modeler", *Cambridge University Press*, 2012, doi: 10.1017/CB09780511667572.

[25] M. Pahlevan and R. Obermaisser, "Evaluation of Time-Triggered Traffic in Time-Sensitive Networks Using the OPNET Simulation Framework", *Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, pp. 283-287, 2018.

[26] D. Basin, C. Cremers, and C. Meadows, "Model Checking Security Protocols", *Handbook of Model Checking*, pp. 727-762, 2018.

[27] S. J. Kim, J. H. Min, and H. N. Kim, "The Development of an IoT-Based Educational Simulator for Dental Radiography", *IEEE Access*, pp. 12476-12483, 2019.