Computers, Materials & Continua
DOI: 10.32604/cmc.2023.028777
Article

Tech Science Press

Check for updates

Application of Physical Unclonable Function for Lightweight Authentication in Internet of Things

Ahmad O. Aseeri[1], Sajjad Hussain Chauhdary[2,*], Mohammed Saeed Alkatheiri[3], Mohammed A. Alqarni[4] and Yu Zhuang[5]

[1]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj, 11942, Saudi Arabia
[2]Department of Computer Science and Artificial Intelligence, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia
[3]Department of Cybersecurity, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia
[4]Department of Software Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia
[5]Department of Computer Science, Texas Tech University, Lubbock, USA
*Corresponding Author: Sajjad Hussain Chauhdary. Email: shussain1@uj.edu.sa
Received: 17 February 2022; Accepted: 07 September 2022

Abstract: IoT devices rely on authentication mechanisms to render secure message exchange. During data transmission, scalability, data integrity, and processing time have been considered challenging aspects for a system constituted by IoT devices. The application of physical unclonable functions (PUFs) ensures secure data transmission among the internet of things (IoT) devices in a simplified network with an efficient time-stamped agreement. This paper proposes a secure, lightweight, cost-efficient reinforcement machine learning framework (SLCR-MLF) to achieve decentralization and security, thus enabling scalability, data integrity, and optimized processing time in IoT devices. PUF has been integrated into SLCR-MLF to improve the security of the cluster head node in the IoT platform during transmission by providing the authentication service for device-to-device communication. An IoT network gathers information of interest from multiple cluster members selected by the proposed framework. In addition, the software-defined secured (SDS) technique is integrated with SLCR-MLF to improve data integrity and optimize processing time in the IoT platform. Simulation analysis shows that the proposed framework outperforms conventional methods regarding the network's lifetime, energy, secured data retrieval rate, and performance ratio. By enabling the proposed framework, number of residual nodes is reduced to 16%, energy consumption is reduced by up to 50%, almost 30% improvement in data retrieval rate, and network lifetime is improved by up to 1000 msec.

Keywords: Cyber-physical systems security; data aggregation; Internet of Things; physical unclonable function; swarm intelligences

1 Introduction

Security is one of the significant challenges for the Internet of Things (IoT) or IoT as they feature more and more frequently in our personal life in the form of myriad applications in diverse sectors such as smart homes, smart transportation, smart city, and smart industrial manufacturing plants [1]. IoT devices face significant issues, including low energy availability [2], a lack of computing resources, and potential cyber-attacks from adversaries [3]. The IoT security triangle comprises three sides, i.e., authentication, authorization, and privacy [4–6]. The first barrier against cyber-attacks is authentication. Authentication services provide a means for the devices on the network to verify the identity of each other [7,8]. Physical unclonable functions (PUFs) have been known for being a lightweight, cost-efficient, and ubiquitous option for serving as the root of trust in an IoT network [9]. PUFs promise a remarkably secure authentication scheme without any cryptographic assets burdening the systems, making them especially interesting for resource-scarce IoT devices [10]. PUF can provide an effective authentication mechanism by providing a device-specific identifier that can form the basis for establishing a secured session key among the devices [11].

A physical unclonable function (PUF) is a function that maps a set of challenges centered on a complex irreproducible physical system to a set of responses. The output of the function can only be accessed through the physical system. Every physical instance of a PUF is unique and cannot be reproduced. In both ICs and FPGAs models, physically unclonable functions (PUFs) are capable primitives for security enhancement [12]. They can be used both for secure key generation and authentication purposes. A PUF can be used to recognize the device on which it is embedded uniquely, and it is improbable to replicate an exact physical unclonable function on other devices due to its unclonable property. By leveraging differences added during the development process, unclonability and uniqueness can be obtained. One of the most common attacks in IoT networks is device tampering and clonability [13]. In the device tampering attack, a secure device on the network is tampered with to make it hostile and compromise the information being transmitted over the network. This tampered device cannot be authenticated if a PUF-based approach is utilized due to the irreproducibility properties of the physically unclonable function. Similarly, a cloning attack attempts to produce a copy of the device that is already authenticated by the system. In this attack, the cloned device acts on behalf of the legitimate device and attempts to eavesdrop or subsequently launch a man-in-the-middle attack. This cloning attack is resisted by a PUF-based authentication system because the cloned device cannot reproduce the exact response characteristic of the original PUF on the authenticated device [14,15].

An ideal PUF cannot be reproduced or cloned. An attempt to clone a PUF will still introduce some minor differences in the manufacturing process, and the response of the copied PUF will differ significantly from that produced by the original PUF. Such differences are random so that devices with unique defects are created and uncontrollable, in the sense that controlling the production system is difficult to replicate them precisely. A physical unclonable function maps a collection of outputs (responses) to the collection of inputs (challenges) in one way, creating a collection of challenge-response pairs (CRPs) that, for each system where the PUF has been applied, is unclonable and unique. In the proposed work, PUF has been employed dynamically to authenticate selected cluster heads, making the resulting IoT system secure against several cryptographic attacks, including man-in-the-middle attacks, cloning attacks, tampering attacks, eavesdropping, etc.

In this paper, we consider an IoT system that consists of numerous devices deployed in a natural environment. Such devices could be interfaced with multiple heterogeneous sensors, such as temperature, humidity, and pressure. IoT devices frequently poll these sensors to gather raw heterogeneous data, eventually transmitted to a central and secure base station. However, these sensors

are battery-operated, making them energy-constrained to not transmit data directly to the base station. For this purpose, we utilize a swarm optimization method, i.e., the ant colony optimization approach, to group the devices into several clusters [16,17]. Each cluster has a dedicated device aggregating information from all the devices within its cluster, referred to as the cluster head (CH). The optimized approach to clustering and selection of cluster heads considers the residual energy, position, velocity, and other characteristics of the devices in a cluster. For the aggregation of information at the cluster head, we utilize the principal component analysis (PCA) approach that reduces the redundancy in the transmitted information and saves transmission and processing energy at the cluster head [18,19]. To ensure the secure operation of the network, we propose that the cluster heads authenticate themselves with the base station using a PUF on the device. That is, the base station sends a challenge to the cluster head's PUF, and the cluster head responds to the authentication request, making the system robust and secure against attacks on the network [20,21]. To efficiently transmit data to the base station, the data from the cluster head must be routed through various relay nodes, which are pre-authenticated by the base station [22–24]. We adopt the software-defined secure (SDS) approach for efficient routing of information to the base station [25,26]. Various devices in an IoT network and their roles are depicted later in Fig. 1. This study is arranged as follows. In Section 1, the significance of security and authentication for IoT networks is emphasized, along with the use of PUFs to provide authentication between the base station and the cluster heads. A thorough literature work has been demonstrated in Section 2. In Section 3, the Secured Lightweight Cost-efficient Reinforcement Machine Learning Framework (SLCR-MLF) is discussed with appropriate mathematical analysis. In Section 4 results, research has been compared based on reliability, accuracy, and performance, and in Section 5, the conclusion and recommendations for future studies have been discussed.

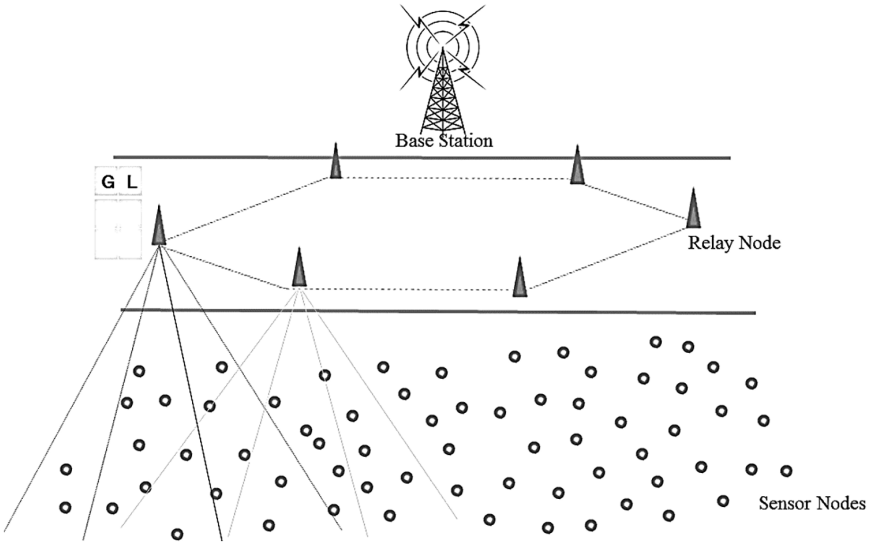

Figure 1: Basic configuration of IoT devices in the system model

2 Literature Review

In this section, we present and discuss recent related works to security considerations and their solutions in the context of software-defined networking employed in IoT systems. To better organize this section, we present the related work in two subsections.

2.1 Secure SDN-Based IoT Networks

Sharma et al. [27] proposed Open cloud software-defined wireless network security for the internet of things (OpCloudSec) security attack mitigation architecture with a highly programmable monitoring network to permit attack identification. The results have shown that the OpCloudSec proposal is successful when it comes to tackling new challenges. The detection algorithm for deduction line packages is quick enough and carries out a high identification rate.

Yan et al. [28] presented a multi-level distributed denial-of-service (DDoS) mitigation framework (MLDMF) to protect against IoT DDoS attacks, including the edge level of computing, the fog level, and the cloud level [29,30]. The networking software specified is used for managing several IoT devices for mitigating DDoS attacks. Experimental findings demonstrate the efficiency of the proposed system for secured data transmission.

Wang et al. [31] analyzed Software-Defined Networking Enhanced Edge Computing (SDNEEC) to determine how SDN and related technologies are incorporated to promote edge server and IoT system management and service [32,33]. They concentrate on how SDN can be used to provide coherent and programmable system management interfaces. The finding of the SDN network is proposed for edge computing, which is explored by the author.

Theodorou et al. [34] proposed a Multi-Protocol Software-Defined Networking (MPSDN) for IoT, which uses the required SDN interface to enforce service knowledge of complex, resource-constrained IoT environments. Several on-demand networking protocols and an Interface provide a tailor-made dashboard and real-time visualization for secured data transmission.

Alqahtani et al. [35] presented a trust-based monitoring security scheme (TBM) to enhance cloud-assisted IoT safety features. This protection framework uses middleware and smart agents to control user and communication safety. The findings show its consistency concerning lower response and detection times, probabilities for misdetection, and false-positive rates. Moreover, a decrease in energy usage has been found to enhance network life for effective data management.

2.2 Recent Authentication Techniques for IoT Networks

Mohanty et al. [36] proposed the Hardware-Assisted Proof-of-Authentication (HA-PoAh) protocol for Data Security on the Internet of Everything (IoE). Proof of PUF-Enabled Authentication mechanism uses the PUF available on each device to reliably produce a specific cloning key, thus ensuring the highest degree of protection. A PUF can create keys using nanoelectronics processing variants that are introduced during the production of an integrated circuit. Therefore, the keys cannot be cloned or produced from any other module once generated with a physical unclonable function module. PUF-based authentication protocols use PUF as the cryptographic primitive and utilize specific cryptographic modules, such as hashing, to achieve secrecy of communication. Also, some IoT-based lightweight authentication schemes have recently been presented in [37–39].

Based on the background research, scalability, device authentication, data integrity, and processing time have been considered IoT devices' challenging characteristics during data transmission. This work presents a new lightweight machine learning-based device authentication technique to enhance security in IoT networks. Our proposed SLCR-MLF has resolved challenging issues related to security and routing efficiency in IoT systems to yield scalable, decentralized, secure, and efficient processing time in IoT devices. In addition, the proposed framework, which internally employs data aggregation along with secure authentication, shows superior performance as compared to OpCloudSec [27],

SDNEEC [31], and MPSDN [34] in terms of energy consumption, data retrieval rate, and network lifetime.

3 SLCR-MLF: Secured Lightweight Cost-Efficient Reinforcement Machine Learning Framework

This section contains the proposed framework, the architecture diagram and coordination, the main system components, explanations of the different algorithms, and performance analysis. Fig. 1 illustrates the basic configuration of the suggested platform containing the sensor nodes, static relay nodes, and base station.

For the IoT device encompassing complex sensors, the energy loss incurred from sensors could be significant. The sensors primarily drop their energy in their motions, interact with other sensors, and conduct actions over the information. In IoT implementations like routine tracking, warning devices and such systems are commonly used. The unique design of the sensors promotes application dynamism while choosing the most fitting node to pick the best energy-efficient route to the base station (BS) in any communication step yields to maximize communication throughputs. This work considers two heuristic methods, the PUF-based authentication mechanism and software-defined system-based routing.

For selecting cluster heads, we consider speed, the location of base stations (BS), the number of neighbors, and the residual energy of the device itself. The proposed system utilizes a swarm-based intelligence methodology for selecting cluster heads where a fitness value is computed each time all swarm members (i.e., sensor nodes), and the node with a better fitness value is chosen as the cluster head (CH). In particular, we employ the ant colony optimization method, a swarm-based optimization technique for solving combinatorial optimization problems, for the cluster heads' election, wherein the fitness functions are calculated at each period, and the one with the better range is selected as the cluster head (CH). The chosen fitness function maintains the complex existence of the system by analyzing the clustering dynamics and removing residual nodes, thus boosting the system's network lifetime.

At the data aggregation stage [40], Principal component analysis is used for dimensionality reduction at the cluster head. Since the devices typically carry heterogeneous sensors, they collect different types of information updated at various rates, hence making transmitting and storing all related information in the network nodes inefficient. PCA solves this issue by providing effective aggregation of data at the cluster heads. Further, PUF provides a basis for authentication among cluster heads and various devices on the IoT network. Since the newly selected node must authenticate with the base station before performing its role, we propose using an authentication mechanism that utilizes challenge-response pairs from a PUF module on the device to create a root of trust with the base station. The techniques mentioned above in the suggested model setup are elaborated below concisely.

In principle, the SDS technique is a computer technology focused on using a given web topology to locate an appropriate path so that data can be transmitted to the base station optimally. When data is ready to be transferred from the cluster heads, it is forwarded to the base station through an efficient routing algorithm. The SDS routing approach selects relay nodes to transmit data to the base station via an optimal route. It also determines the most appropriate node for forwarding at any level. SDS technique is a likelihood-based computational model utilized to determine an optimal pathway in specified network topologies. Factors such as hop count energy and distance from the base station are also considered. Therefore, the most powerful and stable node is selected at all stages, improving the network's lifetime while maintaining low communication latency in the overall system.

As stated earlier, the SLCR-MLF enables gathering heterogeneous sensor data from its multiple cluster members by integrating the SDS technique that enables data integrity while maintaining optimized processing time in the IoT platform. It starts with the sensor nodes being initially deployed at random. The relay nodes are statically deployed around the base station for optimal defense in a hexagonal topology, and the used heterogeneous-nature sensors are distributed randomly, i.e., in various contexts. It is inefficient to bring data from each node to the BS, move it on, and regularly find the most appropriate node to the cluster head (CH). Thus, the CH election is based on its relative distance, velocity, and energy from the relay node.

The base station is the sink utilized for the network propagation and aggregating sensed information from the sensor nodes network. The sensor nodes could monitor anything from ambient conditions, humidity, pressure, temperature, etc. These nodes are distributed randomly to forward data from the sensor network to the base station through the statically allocated relay nodes with higher computing capacity, storage, and coverage. Therefore, the system architecture contains randomly distributed nodes, statically positioned relay nodes, and the base station's registration in a hexagonal structure. The second essential module in the proposed system involves constructing and managing the global and local table, the fitness examination, the cluster head's election, and the clusters' development, along with employing the PUF-based authentication scheme to authenticate devices.

Network topology is the network's spatial structure that defines the nodes' location and how they connect. As seen in Fig. 2, the suggested work prefers a hexagonal topology. The hexagonal topology will increase the coverage so that three adjacent relay nodes are attached to each relay node. Any sensor node will find a relay node at an angle of 120 degrees, meaning that the relay nodes in the device do not exist without a node. The sensor nodes randomly travel around the network to at least one of the relay nodes.

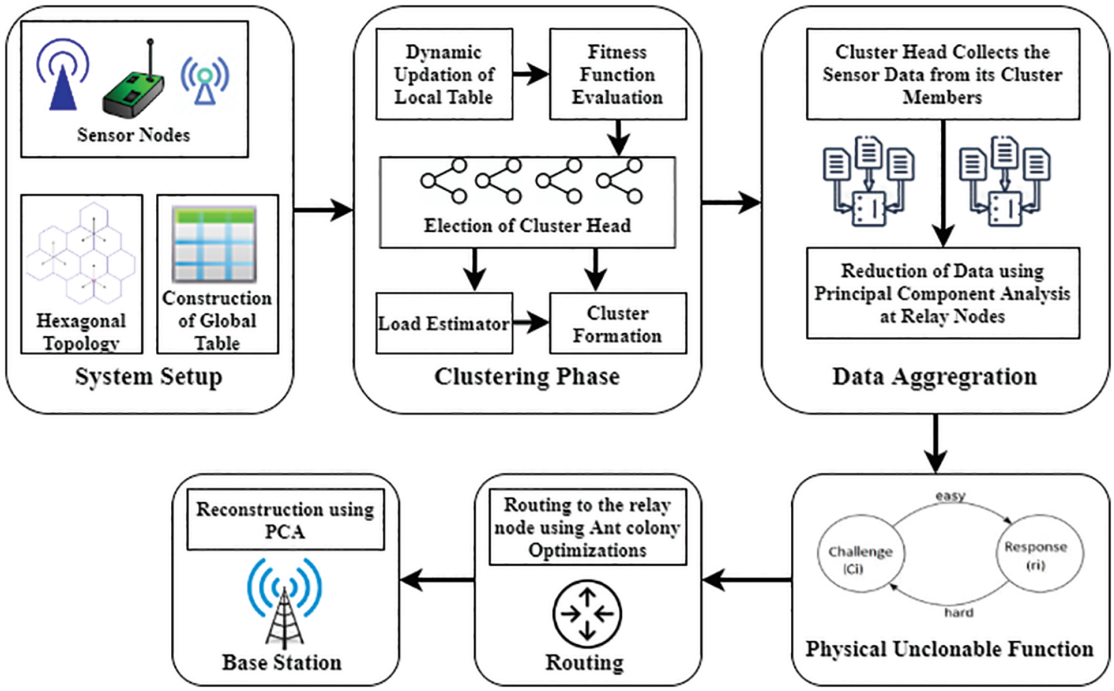

Figure 2: The block diagram of the suggested model

Furthermore, the local table is assisted by any sensor node that is installed within the network. Table 1 displays the local table, including different node parameters such as node id, location, speed, node energy, neighbor count, and list of neighbors. For successful clustering, these parameters are important. In addition, a global table is held at every relay node so that it would be retained to store data about other network relay nodes. As shown in Table 2, the defaults are the node id, global spot (i.e., position), node energy, and hop from the base station. Relay nodes are allowed to broadcast and exchange address messages to the base station. The global table and local table are dynamically altered after frequent breaks. The suggested scheme allows for the complex support of the whole topology. Properly designed systems are needed to have an effective topology control scheme to secure each node and prevent the residual node from being removed from the scheme.

Table 1: Sensor node local table

Node ID	Location	Speed	Node energy	Neighbour's count	Neighbour list

Table 2: Global table of the relay node

Node ID	Position	Hop count	Energy

3.1 Dynamic Clustering

Here, a method to collect data from different sensor nodes is required. Clustering is the process utilized to remove the idleness present in cluster sensor results. For this reason, a cluster head (CH) must be selected first to make the clustering efficient, as it allows for forming related neighbors and the closest distance to the CH nodes at its movement speed. The parameters mentioned above are gathered frequently and stored in a local table from the nodes. Note that each node or particle sensor can become the head of the cluster. Formally, each node can be considered to be an element, and the fitness function $f_{function}$ for each node is evaluated as follows:

$$f_{function} = £_1 a_1 + £_2 a_2 + £_3 a_3 + £_4 a_4 \tag{1}$$

$$a_1 = 1/dR - dP \tag{2}$$

$$a_2 = E_{Avg}\,(C_n) \tag{3}$$

$$a_3 = Number\ of\ neighbor\ nodes \tag{4}$$

$$a_4 = 1/Velocity\ of\ the\ particle \tag{5}$$

Here, the distance of $dR - dP$ among the particles (relay and sensor), the average energy of E_{Avg}, and the number of the clusters is the number of nodes as C_n. As stated in the introduction section, the node with a better fitness value is selected as a CH. The nodes and particles are interchangeably utilized to describe sensor nodes. Since it is complicated, regular table changes, fitness measurements, and prefers CH occur. The other nodes follow the closest Heads to complete the structure of the cluster-based fitness values P_{best} and G_{best}. Since energy is often used for assessing the function, the cluster head does not always consist of the same node. In addition to being ignored, even the lowest-fitness node

is an independent cluster head based on the current and present values $Current_{value} > Present_{value}$. The technique then tends to remove residual node development, as shown in Algorithm 1. The cluster head then delivers the hello note to the nodes in its handling and accepts the acknowledgment messages as a part of the nodes. The framework is the input for this algorithm; the output includes the heads of the cluster. In the fitness function, the velocity parameter is used. In addition, if a node that develops the CH moves at a very significant rate, the grouping must be repeatedly carried out. Given the difficulty and the energy expended on clustering, this is not effective. For CH selection, a node that transfers with a steady speed and does not drastically alter its location and velocity is reflected. The constraint, velocity, is, therefore, inversely relative to the fitness function condition. The cluster head then sends the message to the secured nodes and acknowledges it while the nodes receive acknowledge messages. The device sensors, the output containing the cluster heads, are the inputs to this algorithm. The fitness function requires the velocity parameter, so the clusters must be done repeatedly since the node that is transformed to the cluster head runs fast. Then given the complexity and energy consumed for clustering, this is inefficient. For CH collection, nodes travel progressively and do not radically change their position and velocity.

Algorithm 1: Dynamic clustering pseudocode

Input: Set of nodes S_{nodes} /∗ *nodes have been set in the framework*∗/
Output: Set of cluster heads *CHs* /∗ *Cluster head nodes have been set in the framework*∗/
Begin
if *p* **do**:
set (id, S_{nodes}, P)
calculate ($f_{function} = £_1 a_1 + £_2 a_2 + £_3 a_3 + £_4 a_4$)
calculate P_{best} *and* G_{best}
else if ($current_{value} > present_{value}$) **do**:
update (P_{best})
else
update (G_{best})
End

The cluster head gives a table with the requisite information and selects the next deserving member to be the cluster head when it is about to die or wear out. This means that even if a cluster head fails, we do not lose knowledge from either cluster, thus assuring the system's stability. The method encourages dynamism as new sensors join the cluster, and the table updates the new entrants and measures their residual energy value.

3.2 Data Aggregation in SLCR-MLF

As previously stated, the cluster head needs to gather heterogeneous data from its cluster members in diverse dimensions respective to sensor types, then forwards them to the relay node. It employs a dimensionality reduction algorithm, mainly principal component analysis (PCA), for data aggregation before being forwarded to the base station. The data set is accumulated to a given window frame; the co-variance matrix and mean-variance are measured, and the eigenvectors and eigenvalues are computed. The main factors are extracted, and the reduced data collection is determined using the different measures involved in PCA.

Several key factors identify the degree to which the data is minimized and proportional to the dimensions. The data is expressed in matrices, and dimensions are shortened to determine a matrix

with columns as the key factor. PCA can be beneficial in the suggested heterogeneous scenario since any shape can be compressed into reduced dimensions. This mathematical model is an unsupervised method that decreases the data collection's dimensionality before it is sent, thus minimizing the number of packets that are prerequisites to be safely sent to the base station (BS). PCA is a way of computing the difference in several attributes in a reduced set of variables, thus minimizing the data transmitted to the cluster head (CH) node. The relay node gathers the data before the k values are obtained, which is the size of the window representing the volume of the collected data. Algorithm 2 presents the proposed PCA-based aggregation algorithm. The data aggregation at the relay node matches the PCA in regular mode. The linking records will be transmitted directly from the CH to the BS from the relay nodes in urgent mode.

Algorithm 2: Aggregation of secured transmission

Input: Set S_{nodes} /∗ *nodes have been set in the framework*∗/
Output: Set of *CHs* /∗ *Cluster head nodes have been set in the framework*∗/
Begin
if $S_{nodes} >$ *threshold* **do**:
set (*data, relay_node*)
else:
calculate (Ag_{data})
set Ag_{data} ($data_{size} < Win_{size}$)
forward (Ag_{data})
update (*BS*)
End

The aggregation system is encountered with low and high threshold *th* values for each sensor type. The context-aware framework checks whether the data is within the threshold range while the cluster head gathers data. If not, the base station is alerted of any irregularities and needs immediate transfer based on data size, and window size is denoted as $data_{size} < Win_{size}$. The device would otherwise wait until the window frame obtains the data. The principal components of the proposed data aggregation technique Ag_{data} are shown in Fig. 3. The input is the sensor node data, and the output is the data decreased. A case study to demonstrate the aggregation process is presented in Appendix.

3.3 PUF for Cluster Head Authentication

A PUF-based authentication mechanism is used for authentication with the base station. We assume that a large enough collection of challenge-response pairs has been stored in the BS for each CH candidate device in the IoT network along with its identifier. For authentication, CH sends an authentication request to the BS using its device-specific identifier. Upon receiving the request, the BS looks up a challenge-response pair available in the database corresponding to the received identifier. The challenge is sent to the cluster head, which uses its onboard PUF to compute the response to the received challenge. A hash of this response is then sent back to the BS, which verifies the hash and authenticates the CH device. Let's consider M indicated as y_j with class mean μ_l for class l. The valid samples of every L PUFs form $\cup_l W_l$, whereas the defective samples form $\cup_l C_l$. To neglect defective samples to describe within-class variance as

$$R_S = \frac{1}{|\cup_l W_l| - L} \sum_{l=1}^{L} \sum_{j \in W_l} \left(y_j - \mu W_l\right) \left(y_j - \mu W_l\right)^T \tag{6}$$

satisfying unsamplability needs splitting valid and defective instances of similar PUF and disseminating various valid PUFs separately. Therefore, it has been measured the distance from defective samples, in place of the defective mean, to the valid mean W_l,

$$R_{S_C} = \frac{1}{|\cup_l C_l| - L} \sum_{l=1}^{L} \sum_{j \in W_l} \left(y_j - \mu W_l\right) \left(y_j - \mu W_l\right)^T \tag{7}$$

as inferred from the Eq. (7) where R_{S_C} denotes the class variance. To valid grand mean μ_U yielding in PUFs

$$R_{S_V} = \frac{1}{L} \sum_{l=1}^{L} \sum_{j \in W_l} \left(\mu W_l - \mu_U\right) \left(\mu W_l - \mu_U\right)^T. \tag{8}$$

At the software level, PUF is a one-way function associating a response to a given challenge. Since each PUF's instance possesses its irreproducible properties, PUF enhances the reliability and security tradeoff without reforming the IoT devices. This method guarantees that the information is transmitted through a trustworthy path from the sensor node to the base station.

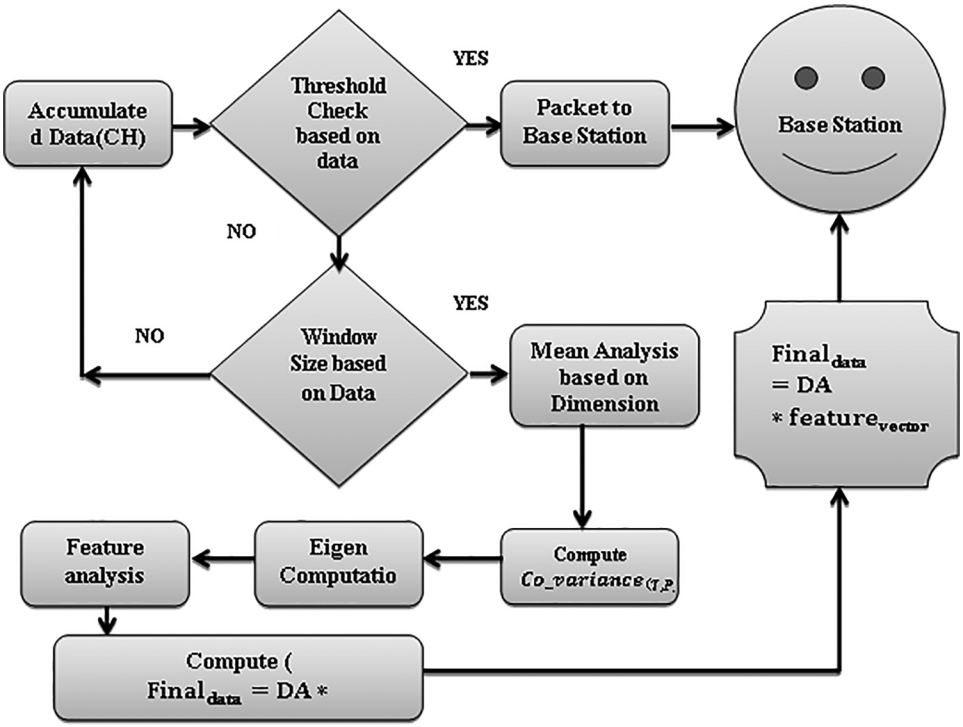

Figure 3: Data is accumulated from multiple sensor nodes. Data aggregation is being performed using PCA at the cluster head. The aggregated data is then transmitted to the base station

3.4 Integration of SDS with SLCR-MLF for Data Integrity

The last step is to transmit data from the relay nodes to the base station. The reconstruction of the PCA-based compressed data is followed by the initial data collection for the data integrity and optimized processing time. The cluster heads serve as an intermediate medium to transfer the detected data through the relay nodes toward the BS, where SDS is used to find the fastest path to the nearest relay node. At each stage, the routing considers the location of the relay node, node energy, and

speed. Starting from a node, the SDS technique specifies an optimized path to the base station through various relay nodes. The pheromone value is determined concerning energy, speed, distance from the node and relay node, and hop count. The minimum hop count node is also taken into consideration for efficient routing.

Algorithm 3: SDS based routing method

Input: Set of particles P /* *particles have been set in the framework* */

Output: Set of pheromones ph /* *pheromones have been set in the framework* */

Begin

if $P_{ph} == Unity$ **do**:

set $(P_{ph} = P_{best})$

else:

set $(P_{ph} = P_{check\_best})$

update (P_{ph})

End

Fig. 4 shows the SDS integrated with the proposed model for data integrity and optimized processing time. The different one-hop neighbors within the CH transmission range will receive the transmission accompanied by each pheromone value. The cluster head then discovers the possibility of each direction across the following equation as

$$g(x,y) = (1-L)\,¥(x,y) + \Delta¥(i,j), \tag{9}$$

where $g(x,y)$ is the pheromone quantity on one particular edge is (x,y), ¥ denotes the pheromone vanishing rate, $\Delta¥(i,j)$ is the pheromone quantity of the formula:

$$\Delta¥(i,j) = \begin{cases} \dfrac{K}{EK}, & if\ ant\ travels\ on\ edge\ (i,j) \\ 0, & otherwise \end{cases} \tag{10}$$

where K is the node's outstanding energy, and EK is the distance between the $hop\_count * base\_station * Velocity$ of the particle and node. A node with the highest pheromone value ph is selected for forwarding of information. Algorithm 3 outlines the algorithm of SDS-based routing.

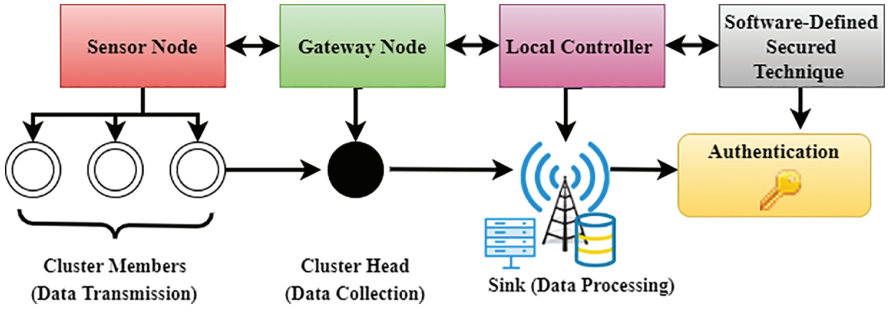

Figure 4: SDS integrated with SLCR-MLF for data integrity, device authentication and optimized processing time

The SDS-based routing is used to check the best probability value of P_{check_best}, taking the likelihood of reaching the destination via an absolute path long before the entire route is taken, thus avoiding the issue of looping over the same path while maintaining secure data transmission. The approach, which

considers the frequency of the path, node energy, and proximity to the relay node, ensures that each idle node's efficiency is not used in transmission, avoiding attacks from malicious nodes and disrupting unreliable or unlicensed nodes.

4 Results and Discussion

In this section, we implement the proposed SLCR-MLF using a network simulator employing OpCloudSec [27], SDNEEC [31], MPSDN [34] to enable the construction of an IoT environment composed of communicating IoT devices. The experiment involves constructing 300 nodes, 6 relay nodes, and a base station. A new method with nodes using Swarm Optimization is implemented to enable dynamic clustering. The energy function used in our methodology helps estimate the dynamism of the node's velocity. The most productive member is selected as the CH, and the remaining node's formation is removed. Fig. 5 displays the map demonstrating the different methods and residual node development. In contrast with other existing methodologies such as OpCloudSec, SDNEEC, and MPSDN, the proposed scheme eliminates the remaining nodes. Even as a node with the least desired fitness value, the remaining nodes' formation becomes a distinct CH in this proposed method and directly sends its data.

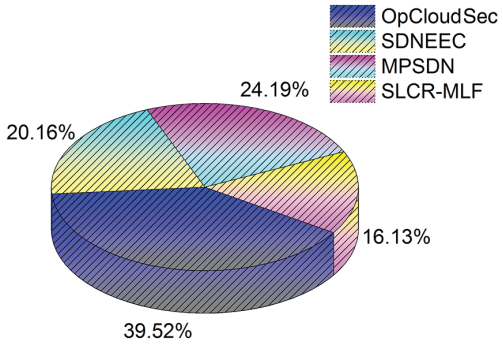

Figure 5: Number of residual nodes with various approaches

Fig. 8 displays the network life after multiple methods have been used. Network lifespan is the period that the whole network is alive or operational. In this case, the diagram is drawn in milliseconds for the duration of the system. The comparison between OpCloudSec, SDNEEC, MPSDN, and the device proposed will be demonstrated. Compared with current approaches, the proposed solution shows an enhanced network life due to proposed algorithms such as dynamic cluster creation, SDS to minimize dimensionality in a heterogeneous setting, and efficiency of data transmission to BS. The node energy is one of the parameters of all proposed algorithms. The above results show that our proposed method improves several performance metrics for the network, i.e., the number of residual nodes is reduced to 16%, energy consumption is reduced up to 50% (Fig. 6), almost 30% improvement in data retrieval rate (Fig. 7), and network lifetime is improved by up to 1000 msec.

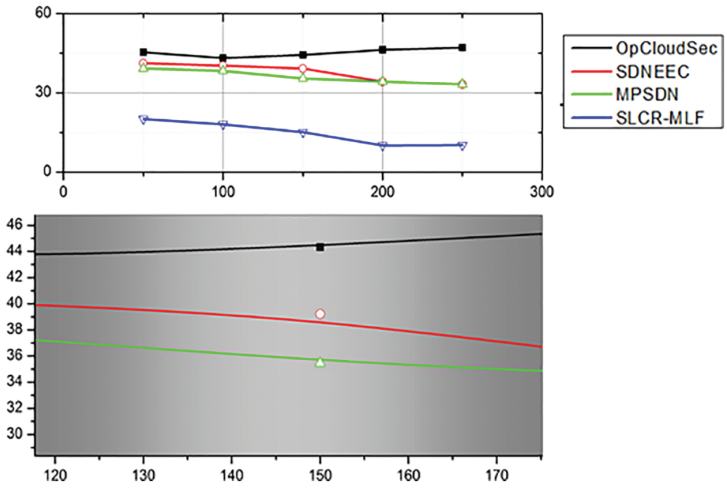

Figure 6: Overall energy consumption at diverse time slots

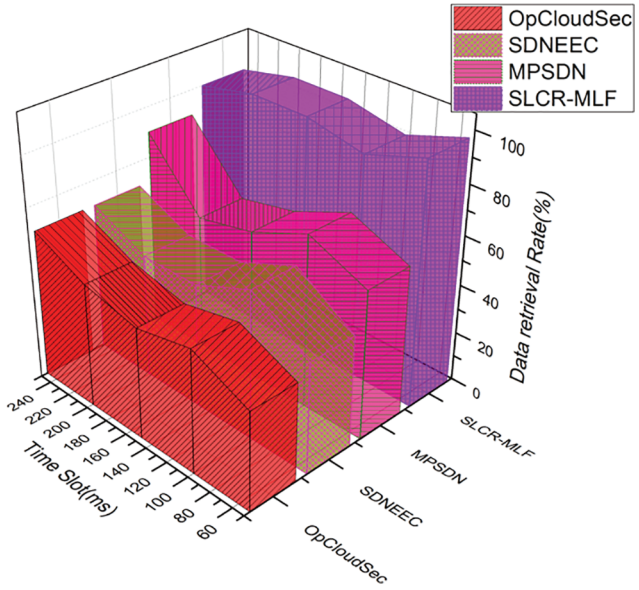

Figure 7: Data retrieval rate at different time slots

Therefore, the proposed SLCR-MLF yields an IoT system that empowers decentralization, scalability, security, data integrity, and optimized processing time in IoT devices. That is, we employ PUF-based authentication to deliver privacy and security requirements within IoT devices, making the system robust to eavesdropping and various cryptographic attacks. At the same time, the proposed SDS technique ensures data integrity and optimized processing time in the IoT platform.

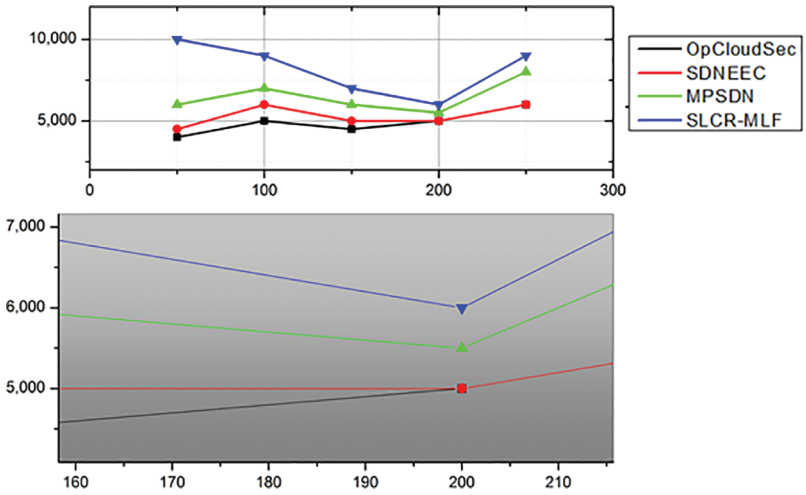

Figure 8: The network lifetime of the various approaches

5 Conclusion and Future Work

A framework for efficient and secure communication between sensor devices in heterogeneous IoT environments has been proposed. The presented cluster formation and selection based on the SLCR-MLF framework yields better performance than the current methods. PCA-based data aggregation allows the integration of various data types, minimizing the size of data and the number of packets transmitted, thus improving energy conservation. The PUF-based authentication mechanism allows cluster heads to authenticate and maintain secure sessions with the base station, making the system robust against various attacks such as man-in-the-middle and cloning attacks. The proposed SDS algorithm guarantees the path's efficiency to the base station, which ensures the selection of the optimal paths with the highest velocity, proximity, and reduced energy usage. Therefore, by reducing the creation of excessive residual nodes, enabling successful authentication of the cluster head via the PUF-based method, and introducing energy-efficient SDS routing, the proposed method enhances connectivity in a highly complex heterogeneous environment. The experimental results show that the proposed SLCR-MLF improves several performance metrics for the IoT network, i.e., the number of residual nodes is reduced to 16%, energy consumption is reduced up to 50%, almost 30% improvement in data retrieval rate, and network lifetime is improved by up to 1000 msec. With the inclusion of greater sensor heterogeneity and more sophisticated PUF designs, the device may be generalized for any complex application to boost the precision of the data recovery process. The proposed approach would require advanced swarming methods in the future.

Funding Statement: The authors received no specific funding for this study.

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

References

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014.

[2] S. C. Shah, A. K. Bashir, S. H. Chauhdary, C. Jiehui and M. Park, "Mobile Ad Hoc computational grid for Low constraint devices," in *Int. Conf. on Future Computer and Communication (ICFCC)*, Kuala Lumpar, Malaysia, pp. 416–420, 2009.
[3] J. Granjal, E. Monteiro and J. Sá Silva, "Security for the internet of things: A survey of existing protocols and open research issues," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1294–1312, 2015.
[4] A. Khanna and S. Kaur, "Evolution of internet of things (IoT) and its significant impact in the field of precision agriculture," *Computers and Electronics in Agriculture*, vol. 157, no. 2, pp. 218–231, 2019.
[5] R. Roman, P. Najera and J. Lopez, "Securing the internet of things," *Computer*, vol. 44, no. 9, pp. 51–58, 2011.
[6] R. Arul, G. Raja, A. O. Almagrabi, M. S. Alkatheiri, S. H. Chauhdary *et al.,* "A quantum-safe key hierarchy and dynamic security association for LTE/SAE in 5G scenario," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 681–690, 2020.
[7] B. D. Deebak and F. Al-Turjman, "Smart mutual authentication protocol for cloud based medical healthcare systems using internet of medical things," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 346–360, 2020.
[8] M. Masud, G. S. Gaba, S. Alqahtani, G. Muhammad, B. B. Gupta *et al.,* "A lightweight and robust secure key establishment protocol for internet of medical things in COVID-19 patients care," *IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15694–15703, 2021.
[9] C. Herder, M. D. Yu, F. Koushanfar and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1126–1141, 2014.
[10] B. Chatterjee, D. Das, S. Maity and S. Sen, "RF-PUF: Enhancing IoT security through authentication of wireless nodes using in-situ machine learning," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 388–398, 2019.
[11] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 580–589, 2019.
[12] Z. Jing, C. Gu, Y. Li, M. Zhang, G. Xu *et al.,* "Security analysis of indistinguishable obfuscation for internet of medical things applications," *Computer Communications*, vol. 161, no. 1, pp. 202–211, 2020.
[13] V. Laguduva, S. A. Islam, S. Aakur, S. Katkoori and R. Karam, "Machine learning based IoT edge node security attack and countermeasures," in *IEEE Computer Society Annual Symp. on VLSI (ISVLSI)*, Miami, Florida, United States, pp. 670–675, 2019.
[14] A. P. Johnson, R. S. Chakraborty and D. Mukhopadhyay, "A PUF-enabled secure architecture for FPGA-based Io applications," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 1, no. 2, pp. 110–122, 2015.
[15] J. R. Wallrabenstein, "Practical and secure IoT device authentication using physical unclonable functions," in *IEEE 4th Int. Conf. on Future Internet of Things and Cloud (FiCloud)*, Vienna, Austria, pp. 99–106, 2016.
[16] D. Liyan, Z. Sainan, T. Geng, L. Yongli and C. Guanyan, "Ant colony clustering algorithm based on swarm intelligence," in *6th Int. Conf. on Intelligent Networks and Intelligent Systems (ICINIS)*, Shenyang, China, pp. 123–126, 2013.
[17] R. S. Parpinelli, H. S. Lopes and A. A. Freitas, "Data mining with an ant colony optimization algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 321–332, 2002.
[18] Z. Jellali, L. N. Atallah and S. Cherif, "Principal component analysis based clustering approach for WSN with locally uniformly correlated data," in *15th Int. Wireless Communications & Mobile Computing Conf. (IWCMC)*, Tangier, Morocco, pp. 174–179, 2019.
[19] X. Zhang, H. Wu, Q. Li and B. Pan, "An event-based data aggregation scheme using PCA and SVR for WSNs," in *IEEE 85th Vehicular Technology Conf. (VTC Spring)*, Sydney, NSW, Australia, pp. 1–5, 2017.
[20] A. Mars and W. Adi, "Clone-resistant entities for vehicular security," in *Int. Conf. on Innovations in Information Technology (IIT)*, Al Ain, United Arab Emirates, pp. 18–23, 2018.
[21] K. Yang, K. Zheng, Y. Guo and D. Wei, "PUF-based node mutual authentication scheme for delay tolerant mobile sensor network," in *7th Int. Conf. on Wireless Communications, Networking and Mobile Computing*, Wuhan, China, pp. 1–4, 2011.

[22] J. A. Kim, D. G. Park and J. Jeong, "Design and performance evaluation of cost-effective function-distributed mobility management scheme for software-defined smart factory networking," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 6, pp. 2291–2307, 2020.
[23] H. Babbar and S. Rani, "Software-defined networking framework securing internet of things," in *Integration of WSN and IoT for Smart Cities*, Chapter no. 1, Springer, Cham, Switzerland, pp. 1–14, 2020.
[24] I. A. Jovan, K. Sharif, F. Li, Z. Latif, M. M. Karim *et al.*, "A survey of network virtualization techniques for internet of things using SDN and NFV," *ACM Computing Surveys (CSUR)*, vol. 53, no. 2, pp. 1–40, 2020.
[25] K. Kaur, S. Garg, G. S. Aujla, N. Kumar, J. J. Rodrigues *et al.*, "Edge computing in the industrial internet of things environment: Software-defined-networks-based edge-cloud interplay," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 44–51, 2018.
[26] A. Bradai, M. H. Rehmani, I. Haque, M. Nogueira and S. H. Bukhari, "Software-defined networking (SDN) and network function virtualization (NFV) for a hyperconnected world: Challenges, applications, and major advancements," *Journal of Network and Systems Management*, vol. 28, no. 3, pp. 433–435, 2020.
[27] P. K. Sharma, S. Singh and J. H. Singh, "Opcloudsec: Open cloud software defined wireless network security for the internet of things," *Computer Communications*, vol. 122, pp. 1–8, 2018.
[28] Q. Yan, W. Huang, X. Luo, Q. Gong and F. R. Yu, "A multi-level DDoS mitigation framework for the industrial internet of things," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 30–36, 2018.
[29] R. Doshi, N. Apthorpe and N. Feamster, "Machine learning DDoS detection for consumer internet of things devices," in *IEEE Security and Privacy Workshops (SPW)*, San Francisco, CA, USA, IEEE, pp. 29–35, 2018.
[30] D. Yin, L. Zhang and K. Yang, "A DDoS attack detection and mitigation with software-defined internet of things framework," *IEEE Access*, vol. 6, pp. 24694–24705, 2018.
[31] A. Wang, Z. Zha, Y. Guo and S. Chen, "Software-defined networking enhanced edge computing: A network-centric survey," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1500–1519, 2019.
[32] J. Ni, X. Lin and X. S. Shen, "Toward edge-assisted internet of things: From security and efficiency perspectives," *IEEE Network*, vol. 33, no. 2, pp. 50–57, 2019.
[33] G. Manogaran, C. Thota and M. V. Kumar, "Meta cloud data storage architecture for big data security in cloud computing," *Procedia Computer Science*, vol. 87, pp. 128–133, 2016.
[34] T. Theodorou, G. Violettas, P. Valsamas, S. Petridou and L. Mamatas, "A multi-protocol software-defined networking solution for the internet of things," *IEEE Communications Magazine*, vol. 57, no. 10, pp. 42–48, 2019.
[35] F. Alqahtani, Z. Al-Makhadmeh, A. Tolba and O. Said, "TBM: A trust-based monitoring security scheme to improve the service authentication in the internet of things communications," *Computer Communications*, vol. 150, no. C, pp. 216–225, 2020.
[36] S. P. Mohanty, V. P. Yanambaka, E. Kougianos and D. Puthal, "PUFchain: A hardware-assisted blockchain for sustainable simultaneous device and data security in the internet of everything (IoE)," *IEEE Consumer Electronics Magazine*, vol. 9, no. 2, pp. 8–16, 2020.
[37] A. Diro, H. Reda, N. Chilamkurti, A. Mahmood, N. Zaman *et al.,* "Lightweight authenticated-encryption scheme for internet of things based on publish-subscribe communication," *IEEE Access*, vol. 8, pp. 60539–60551, 2020.
[38] M. Almulhim and N. Zaman, "Proposing secure and lightweight authentication scheme for IoT based E-health applications," in *IEEE 20th Int. Conf. on Advanced Communication Technology (ICACT)*, Chuncheon, South Korea, pp. 481–487, 2018.
[39] Y. J. Ren, Y. Leng, J. Qi, P. K. Sharma, J. Wang *et al.,* "Multiple cloud storage mechanism based on blockchain in smart homes," *Future Generation Computer Systems*, vol. 115, no. 2, pp. 304–313, 2021.
[40] H. J. Lee, M. T. Soe, S. H. Chauhdary, S. Rhee and M. S. Park, "A data aggregation scheme for boundary detection and tracking of continuous objects in WSN," *Intelligent Automation & Soft Computing*, vol. 23, no. 1, pp. 135–147, 2016.

Appendix A

The elected Cluster Leader carries out data aggregation. The cluster head continues to gather heterogeneous sensor information from its multiple cluster members. The accumulated heterogeneous CH information is sent to the relay node. It executes the PCA to combine heterogeneous information into a minimal data set and forward the compact data into the base station. Suppose three sensor types exist: temperature, pressure, and humidity. PCA allows the storage of multi-dimensional data. The PCA steps and sample data are given below. The CH first gathers and creates a data matrix from heterogeneous sensors. Let us presume that there are three categories of sensors in the cluster: humidity, pressure, and temperature; in each group, three sensors. The constructed 3×3 data matrix is the same as the one below.

Temperature (T)	*Pressure* (P)	*Humidity* (H)
25	130	15
31	134	16
32	133	17

Corollary 1

The next step is to measure each dimension's mean. Subtract from its mean each value of a specific dimension, i.e., $\left(\mathrm{a}-\overline{\mathrm{A}}\right), \left(\mathrm{b}-\overline{\mathrm{B}}\right) \wedge \left(\mathrm{c}-\overline{\mathrm{C}}\right)$ build the Data Adjust matrix.

$$|DA| = \begin{bmatrix} -3.33 & -2.60 & -3.30 \\ -0.43 & 2.30 & -0.30 \\ 3.66 & 0.43 & 3.66 \end{bmatrix}$$

Corollary 2

The following step will be to define the co-variance of the data adjustment matrix assumed with the co-variance Eq. (A.1):

$$Co\_variance_{(T,P,H)} = \begin{bmatrix} cov\,(t,t) & cov\,(t,p) & cov\,(t,h) \\ cov\,(p,t) & cov\,(p,p) & cov\,(p,h) \\ cov\,(h,t) & cov\,(h,p) & cov\,(h,h) \end{bmatrix} = \begin{bmatrix} 6.66 & 3.77 & 6.66 \\ 3.77 & 6.66 & 3.77 \\ 6.66 & 3.77 & 6.66 \end{bmatrix}.$$

As inferred from the above matrices, where the co-variance is of the pressure and temperature, i.e., $Co\_variance_{(T,P)}$ is calculated utilizing the following equation:

$$Co\_variance_{(T,P)} = \frac{\sum_{i=1}^{n} (t_i - T)\,(p_i - P)}{n-1}. \tag{A-1}$$

Then, find the co-variance matrix $\sum_{i=1}^{n} (t_i - T)\,(p_i - P)$. The eigenvector is a matrix that gives the matrix of a new form if multiplied by the ideal matrix $n-1$. This matrix can be articulated as an Eigenvector scalar product. The word scalar is referred to as the value of Eigen.

Corollary 3

Let matrix $\mathrm{A} = Co\_variance_{(T,P,H)}$, the Eigenvalues of matrix A is determined as follows:

$$A = \begin{bmatrix} 6.66 & 3.77 & 6.66 \\ 3.77 & 6.66 & 3.77 \\ 6.66 & 3.77 & 6.66 \end{bmatrix}$$

$$|A - \lambda I| = 0$$

By solving the above equation, we get Eigenvalues $\lambda_1 \cong 15.1$, $\lambda_2 \cong 5.0$ *and* $\lambda_3 \cong 3.1 \times 10^{-14}$.

Then, measure the function's vector and reduce the size as the key variable is picked the eigenvector with the largest Eigenvalue (15.1). A function vector is generated in the corresponding line in the eigenvector. The higher the dimension, the smaller the loss of compression. So, this is the functional vector.

$$feature_{\text{vector}} = \begin{bmatrix} -0.72 \\ -0.56 \\ -0.72 \end{bmatrix}$$

The final condensed data is calculated utilizing the following Eq. (A.2):

$$Final_{\text{data}} = DA * feature_{\text{vector}} \tag{A-2}$$

$$Final_{\text{data}} = \begin{bmatrix} -3.33 & -2.60 & -3.30 \\ -0.43 & 2.30 & -0.30 \\ 3.66 & 0.43 & 3.66 \end{bmatrix} * \begin{bmatrix} -0.72 \\ -0.56 \\ -0.72 \end{bmatrix} = \begin{bmatrix} 4.09 \\ -0.77 \\ -5.5 \end{bmatrix}.$$