

Original Research

Received 05 June 2026

Revised 22 July 2026

Accepted 05 August 2026

Natural Resources for Human Health 2026, Vol. 6 (14s): 768–790

KEYWORDS:

Silver nanoparticles; Green synthesis; Medicinal plants; Antibacterial activity; Cytotoxicity; Nanomedicine.

Ontology-Driven Attribute-Based Access Control with Optimized Martino Encryption for Secure Healthcare Cloud Data

Suthan Renuka^{1*}, Chirathally Dyamanna Guruprakash², Srinidhi G A³

¹Research Scholar, Department of Computer Science and Engineering, Sri Siddhartha Institute of Technology, Sri Siddhartha Academy of Higher Education, Maralur, Tumakuru, Karnataka – 572105, India, sutrenuka123@gmail.com

²Department of Computer Science and Engineering, Shridevi Institute of Engineering and Technology, Tumakuru, India, sutrenuka123@gmail.com

³Professor, Sri Siddhartha Institute of Technology, Sri Siddhartha Academy of Higher Education, Maralur, Tumakuru, Karnataka – 572105, India

³Associate Professor, Department of CSE(Cyber Security) Sri Siddhartha Institute of Technology, SSAHE, Tumakuru

ABSTRACT: Ontologies are extensively utilized in various domains to model trust-based data access management. However, enforcing time-limited access permissions remains difficult and may lead to unauthorized access after the collaboration period expires. Initially, the Kohonen Self-Organizing Neural Network (KSONN) analyzes healthcare requests and classifies them as normal or malicious. Normal requests are then scheduled using the Prioritized Sorted Task Scheduling Algorithm (PSTSA), with task priority determined from the request context rather than from the user role alone. The fuzzy system considers the role, urgency, clinical criticality, impact, resource availability, and deadline proximity. Ontology-derived facts are mapped to eXtensible Access Control Markup Language (XACML) subject, resource, action, and environment attributes before policy evaluation. For example, ontology reasoning identifies the role of the user and the relationship to a patient record, and these values are supplied as subject and resource attributes to the XACML policy decision point. Cluster labels are assigned using only the class labels of the training samples mapped to each cluster. Test-set labels are not used during cluster formation or cluster-label assignment and are only used for the final performance evaluation. If the data is classified as normal, the job scheduling process is performed based on the PSTSA. Finally, patient privacy is protected using the proposed cryptographic method (MESBO-based Martino encryption), and the encrypted healthcare data are securely stored in the cloud. The framework is estimated using an 80:20 train-test split, and the reported results represent the average performance across ten independent experimental runs. The representative KSONN test run achieved 97.93% accuracy and 99.39% precision, whereas the ten-run mean performance is used for comparative evaluation.

1. INTRODUCTION

Digital records that are electronically saved and hold all of the medical data of a patient require measures to avert and address serious risks that may jeopardize the well-being of individuals beyond national boundaries. Protecting private patient information and complying with healthcare data security regulations are critical requirements in the healthcare sector [1]. The

field of medical information has advanced significantly with the emergence of information technologies [2]. Electronic Health Records (EHRs) are computerized records that contain comprehensive data about the health history, diagnoses, lab results, treatments, prescriptions, and other medically relevant information of patients. They allow for easy management of healthcare services with respect to data confidentiality and rapid access to patient medical information for use by approved healthcare practitioners. The adoption of secure healthcare data can greatly enhance patient safety and aid healthcare standards [3]. Hospital systems are the primary setting in which medical data are managed and stored when Electronic Health Record (EHR) systems are initially developed using centralized databases. Additionally, different healthcare providers exchange and manage patient data through Electronic Health Record (EHR) systems, enabling authorized healthcare professionals to securely access, update, and share patient information across healthcare facilities [4].

Machine Learning (ML) algorithms such as K-Nearest Neighbor (KNN), Artificial Neural Network (ANN), and Support Vector Machine (SVM) are utilized for detecting threats in healthcare data. Moreover, these techniques have limited capability to preserve the topological relationships among healthcare network traffic patterns. Hence, the Kohonen Self-Organizing Neural Network (KSONN) is employed in this framework. Furthermore, integrating ML with Internet of Things (IoT) devices simplifies the monitoring, management, and analysis of medical data. Additionally, ML algorithms can efficiently process vast amounts of biological data. However, existing decentralized access control methods cannot ensure the transparency and security of access control policies because these policies are stored in the cloud. Healthcare tasks are prioritized using urgency, criticality, deadline, and resource requirements rather than execution duration alone [5]. Healthcare data security issues have been addressed through ontology-based interoperability, attribute-based access control, role-based access control, and context-aware access control [6].

Role-Based Access Control (RBAC) is an organized approach to enhancing healthcare data security by assigning access rights according to predefined user roles, and it has attracted significant scholarly attention [7]. Consequently, further investigation is required to guarantee a strong and efficient application of ABAC for security enhancement [9]. This includes a lack of comprehensive assessments, oversight of potential security and privacy concerns, insufficient technical implementation details, and the absence of evaluations from real-world deployments [10]. Temporal and attribute-based access-control mechanisms have also been investigated to strengthen healthcare data security [11]. To address these issues, an attribute-based access control protocol combined with an optimized encryption algorithm has been developed to enhance the security of healthcare data [12]. Therefore, every distinct module in the proposed framework performs a diverse task with a specific function related to security. Moreover, the KSONN is used to pinpoint malicious activities, and the Prioritized Sorted Task Scheduling Algorithm (PSTSA) is intended to prioritize health-related tasks according to their urgency and execution requirements within the healthcare cloud environment. RBF is assigned the job of task prioritization; XACML is responsible for granting access control and Max- value Entropy Search Bayesian Optimization optimizes the cryptographic key-generation process by maximizing key randomness and entropy while minimizing key correlation and repetition. MESBO optimizes the cryptographic key-generation process by maximizing key randomness and entropy while minimizing key correlation and repetition. Ontology-based log monitoring has also been investigated to improve security observability and incident response in cloud environments [21]. The XACML module is evaluated in terms of policy decision latency, request throughput, scalability with increasing policy complexity, and conflict resolution behavior. The proposed MESBO algorithm enhances security and key generation efficiency compared with existing optimization approaches by producing highly random and optimized encryption keys.

The proposed approach combines multiple access control mechanisms used in healthcare settings to provide more comprehensive and fine-grained security. RBAC provides role-based permissions, CAAC incorporates contextual information into access decisions, and ABAC enables fine-grained authorization based on multiple attributes. Their integration helps address the limitations of using these mechanisms independently and improves access flexibility and security in healthcare settings. These access control frameworks consider multiple mechanisms, including role-based, context-aware, and attribute-based access control; however, they use them independently. The new proposed framework is innovative because (i) it combines a KSONN for attack identification with ontology-driven access management, (ii) it utilizes RBF to help prioritize healthcare stakeholders' task, (iii) it employs XACML to generate more fine-grained authorization decisions, and (iv) it uses SBO to create an optimal key generation solution for Martino encryption. All of these contributions work together to enhance confidentiality, access flexibility, and computational efficiency for cloud computing in the field of healthcare systems. To enhance fine-grained

attribute-based authorization, the proposed framework adopts XACML, which estimates ontology-derived access policies before healthcare resources are accessed.

The key contributions of the proposed framework are as follows:

- An optimized encryption algorithm and an attribute-based access control scheme have been developed to enhance the security of healthcare data.
- KSONN is used for attack detection by categorizing each request as normal or attack.
- Job scheduling is performed by assigning legitimate healthcare tasks according to their priority using the PSTSA technique, ensuring that urgent and critical healthcare requests are processed efficiently while optimizing available cloud resources. Each healthcare request is assigned one of three priority levels: low, medium, or high, based on the RBF scheduling mechanism.
- XACML is developed to prevent unauthorized users from accessing data, and based on the access control policy, it evaluates the authorization of a user to view medical documents based on specific attributes.
- A MESBO-based cryptographic algorithm is proposed to enhance the confidentiality and secure storage of healthcare data in cloud environments.

The organization of this paper is organized as follows: Section 2 reviews the literature on ontology-driven healthcare security and attribute-based access control. Section 3 presents the proposed MESBO-XACML framework and its methodology. Section 4 discusses the experimental setup, performance evaluation, and comparative analysis. Finally, Section 5 concludes the paper and presents the future research directions.

1. LITERATURE REVIEW

Ontology-oriented security enhancement has been widely adopted to improve secure healthcare data sharing, access control, and privacy preservation in cloud-based healthcare systems. Zenghui Yang et al. [13] developed a blockchain technology-based access control scheme on attributes specifically designed for the protection of IoT data. Initially, data such as location, temperature, heart rate, and BP were collected using IoT devices, namely sensors, smartwatches, wearables, and smart phones. Thereafter, these data were divided into packets, and an ID was generated by the IoT gateway to uniquely identify healthcare devices before secure communication and data transmission. Then, this gateway was used as a blockchain node to initialize encryption by calling smart contracts. Subsequently, the encryption algorithm produced ciphertext, which was securely transmitted and stored in the cloud for authorized access. Therefore, this model enhanced access-control security, ownership verification, and protection against impersonation. Therefore, this model enhanced policy hiding and ownership verification and eliminated impersonation. However, the low storage capacity and computational power of IoT devices affected the overall performance of the model.

Xue Zhang et al. [14] introduced a TDAC using semantic inference technologies for logistics businesses. Initially, this model constructed an ontology for knowledge modeling called Logistics Business Collaboration Ontology (LBCO) by combining logistics domain and access control ontologies. Subsequently, a semantic access policy was set to improve the performance of the Semantic Web Rule Language (SWRL), which was used to allow authorized access. Finally, the access control policies were optimized to make the rules clear and efficient. Therefore, this model enhanced semantic interoperability, access control, and authorization. However, the proposed approach suffered from limited rule readability and poor policy-authoring usability, which reduced its effectiveness in practical healthcare environments.

Binyong Li et al. [15] developed a Context-Aware Risk Attribute Access Control (CARAC) a risk indicator system to evaluate risk levels and make access decisions. First, a risk assessment factor system was employed to calculate the risk of allowing user access. Thereafter, the CARAC adapted game-theoretic combination weighting to dynamically evaluate risk levels and enhance accuracy and reliability. The model then allocates risk levels to access requests based on user identity, behavioral data, and environmental factors. Therefore, the model demonstrated fine grained authorization, but its unoptimized risk perception negatively affected prediction precision.

Rudri Kalaria et al. [16] introduced a fog computing model called Adaptive Context-Aware Access Control (ACAAC). Initially, this model implemented a Fog-Based ACAAC (FB-ACAAC) model to discard unauthorized access adjustment based on related

data. Then, authentication and access control were performed to ensure security and trustworthiness. Thereafter, managing context information included context modeling, data collection, and context distribution. Therefore, this model significantly reduced information processing time while improving resource utilization and overall system efficiency. Furthermore, FB-ACAAC eliminated communication overhead by pushing components closer to IoT devices. However, FB-ACAAC limits the user privileges by enforcing fine-grained attribute-based access policies while ensuring secure and authorized access to healthcare resources.

M. Pavithra et al. [17] demonstrated Boneh Goh Nissim Bilinear Attribute-Based Optimal Cache Oblivious (BGNBA-OCO), a privacy-preserving attribute-based access control model. First, this model used deduplication to protect sensitive information in the cloud by assigning public keys to patient files. OCO was then used for data deduplication verification, which reduced the communication overhead. Subsequently, the processed file underwent authorization and decryption processes to ensure secure access and protect the information from unauthorized users. Therefore, only authorized users were allowed to access these files, thus maintaining confidentiality and integrity. However, the use of OCO results in increased storage consumption and complexity.

Malte Kerl et al. [18] developed an optimized Zero-Knowledge Proof (ZKP) by reviewing Homomorphic Encryption (HE) and Attribute-Based Access Control (ABAC) for privacy preservation. First, this model obtained information from the IoT network, and then the ZKP communicated with entities that possess the private key. Then, attribute data collection was separated from the access control decision protocol to allow the system to work in a decentralized manner, as mines depend on high availability. Therefore, cyber-physical systems can only access the environmental context of the mine. However, attribute disclosure to unwanted personal information was possible through adversarial policies, which affect privacy.

In general, existing research has been discussed individually in terms of ontology-based access control, context-aware authorization, optimization, and encryption. However, they do not propose a unified solution that combines intelligent intrusion detection, ontology reasoning, attribute-based authorization, task scheduling, and optimized encryption, all of which are important for protecting healthcare sites from cyberattacks. This research addresses these limitations using the MESBO-XACML framework.

2. PROPOSED METHODOLOGY

The main aim of this research is to develop an ontology-based healthcare security framework that improves healthcare data protection. The main objective of access control is to prevent unauthorized people from reading or changing private information. Securing these communications and ensuring that people providing information use appropriate digital channels can be challenging. To overcome these issues, an attribute-based access control protocol combined with an optimized encryption algorithm has been established. The complete workflow of the proposed framework is illustrated in Figure 1.

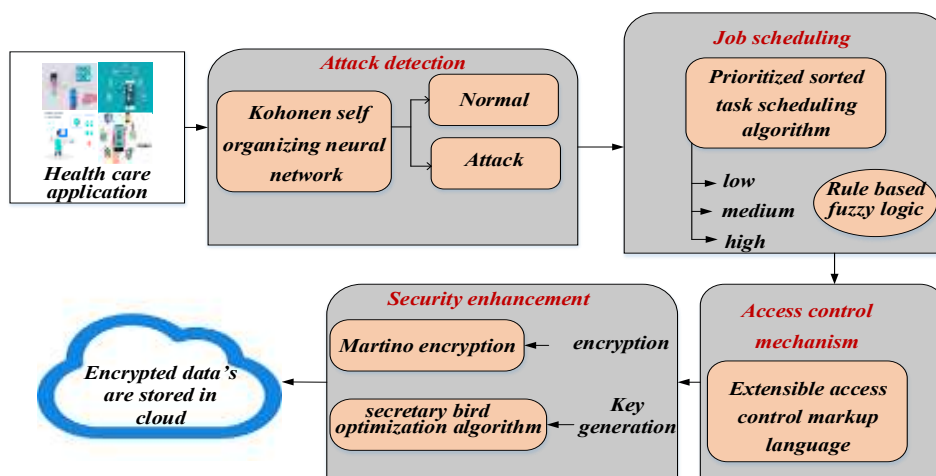


Figure 1. Architecture of proposed MESBO-XACML framework for ontology-based healthcare data security.

The ontology reasoning engine estimates the semantic relationships among healthcare users, roles, permissions, and resources before forwarding validated authorization requests to the XACML policy decision point for access evaluation. The encryption configuration specifies the key length, initialization vector or nonce generation method, block-processing mode, padding scheme, and authentication mechanism required for secure encryption and decryption. During emergencies, authorized healthcare professionals temporarily access controlled patient records using an emergency access mechanism, with each access recorded in audit logs for accountability. The MESBO-based Martino encryption procedure consists of key generation, binary transformation, key-dependent matrix rotation, permutation, and inverse operations for decryption. KSONN is implemented as an unsupervised self-organizing map for clustering healthcare network requests. After training, each cluster is assigned a normal or attack label according to the majority class of the training samples that are mapped to that cluster. The resulting cluster labels are then used to classify the unseen test requests. The KSONN receives the extracted healthcare network traffic features and produces a normal/attack classification. Normal requests are forwarded to the PSTSA module, which receives the task attributes and produces a task execution order. The contribution of the fuzzy-priority module should be evaluated by comparing the proposed fuzzy prioritization scheme with a non-fuzzy baseline under the same healthcare task workload. The ontology and XACML modules receive user, resource, action, and environmental attributes and produce permit or deny authorization decisions. Authorized healthcare data are subsequently processed by the MESBO-based encryption module, which generates an encryption key and produces a ciphertext for secure storage. The ontology performance is evaluated using the ontology size, reasoning time, consistency verification, and correctness of the inferred access-control relationships. In this phase, the process of assigning task priority is accomplished using Rule-Based Fuzzy Logic (RFL). Secretary Bird Optimization (SBO) is a population-based metaheuristic optimization algorithm consisting of exploration and exploitation phases [22]. After completion of this process, XACML checks user authorization for access rights using the ontology access policy. Finally, the encrypted healthcare data are securely stored in cloud using the proposed MESBO-based Martino encryption algorithm. The proposed ontology model highlights the connections between healthcare stakeholders, such as patients, doctors, nurses, management, medical records, and others, relating to roles, privileges, and other resources. All the discussed entities are defined using Ontology Web Language concepts, whereas the relationships between diverse roles are shown by the utilization of the Resource Description Framework. Therefore, the ontology is established by employing the Protégé, while semantic reasoning is performed using the pellet reasoning engine to infer relationships between the healthcare entities and access permissions. During access control, ontology reasoning verifies the semantic relationships between users, roles, healthcare resources, and permissions before forwarding authorization requests to XACML policy engine for final access decisions. If data are normal, the process moves on to job scheduling, where tasks are assigned based on user-defined priorities. The PSTSA performance is evaluated using the waiting time, turnaround time, makespan, task rejection rate, and deadline-miss rate. Task are assigned to users based on their respective types and requirements, ensuring that each task is appropriately allocated to the most suitable users for efficient execution and improved resource utilization. Based on RFL, each healthcare request is assigned one of three priority levels—low, medium, or high—according to its urgency, clinical impact, patient criticality, resource availability, and deadline proximity. This prioritization ensures that urgent and clinically critical healthcare tasks are processed with higher priority. Emergency access is granted only under predefined emergency conditions and for a limited duration. Each emergency access event is recorded in the audit log, and the corresponding access privilege is revoked after the defined emergency period. Notifications and subsequent audit reviews are used to maintain accountability. The XACML access control policy, which is attribute-based, determines who is authorized to receive medical information. Finally, a Martino encryption technique based on MESBO cryptography is developed to protect patient privacy and enhance the security of the contents of the database. The proposed encryption mechanism primarily addresses the issue of data confidentiality. Cryptographic evaluation includes ciphertext correlation and byte-level entropy, and additional randomness, key sensitivity, and authenticated decryption tests are required for a more comprehensive security assessment. The evaluated KSONN module focuses on distinguishing between normal and malicious healthcare network traffic. Additional threats, including replay attacks, insider misuse, and key compromise attempts, are considered at the framework design level but are not independently evaluated in the present experiment. XACML restricts access according to defined attributes and policies; however, authorization alone does not prevent misuse by authorized users. Therefore, audit logging and least-privilege policies are required to mitigate insider misuse.

2.1. Attack Detection

Attack detection is a system developed to detect and mitigate attacks in communication networks. It employs various techniques, including signal-based detection, packet-based analysis, proactive algorithms, and hybrid strategies, to safeguard against threats. In this paper, KSONN is utilized for effective attack detection by identifying and classifying malicious activities, by improving overall security and reliability of system. If an incoming request is classified as malicious, it is immediately blocked, logged, and isolated from the healthcare system. Only legitimate and authenticated requests are processed for task scheduling, access control, and encryption. Each access request contains a timestamp and a unique nonce to ensure request freshness, prevent replay attacks, and enhance the overall security of the access control process the authorization module verifies the freshness and uniqueness of these values before processing the request, thereby preventing the reuse of previously accepted requests. Finally, encryption techniques are employed to ensure data confidentiality and protect sensitive information from external hackers and other unauthorized users, thereby enhancing complete security of system. Replay protection is considered a security requirement of framework but is not independently implemented or evaluated in the current prototype.

2.1.1. Kohonen Self-Organizing Neural Network

The KSONN architecture consists of an input layer, an output layer, and a weight vector connecting the two layers, with the output layer also incorporating a neighborhood structure. There are 11 security features that consist of several parameter including source IP, destination IP, protocol type, flow duration, packet rate, connection status, service type, transmission interval, payload size, and traffic behavior, which should be normalized prior to clustering for better detection results. The architecture of KSONN is described below, highlighting its key components overall functionality. Figure 2 illustrates the proposed KSONN framework and the flow of data through KSONN.

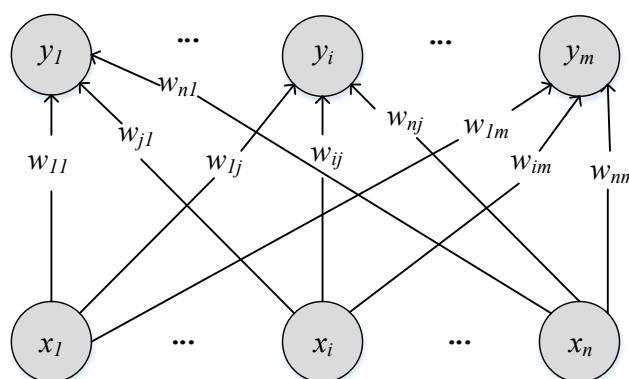


Figure 2. Framework for KSONN

The KSONN module analyzes healthcare IoT network traffic by extracting relevant security features and clustering the data into normal and attack categories for further access control decisions. The categorical features are encoded before numerical scaling and then the scaling parameters are estimated from the training set to ensure that the data are transformed using the same scaling parameters and to prevent information leakage.

Kohonen learning is a self-organizing, unsupervised process, which makes it unsuitable for direct application in forecasting tasks. However, the proposed KSONN model addresses this limitation by clustering healthcare requests based on feature similarity, thereby improving attack detection accuracy and reducing false classifications.

The KSONN is trained using the training samples and their corresponding class labels. For each input feature vector x_i , the best matching neuron c is identified by minimizing the Euclidean distance between x_i and the neuron weight vectors w_j . The winning neuron is then updated using the reward-and-punishment learning rule given in Eq. (1). The target value for x to s is set and the actual output of neuron c is denoted as o . Then, the reward and punishment formula is applied to finalize the process, which is described as Eq. (1).

$$\begin{cases} w_c(t+1) = w_c(t) + \eta(t) * (x_i(t) - w_c(t)), & f_{abs}(s-0) < e \\ w_c(t+1) = w_c(t) - \eta(t) * (x_i(t) - w_c(t)), & f_{abs}(s-0) \geq e \\ w_i(t+1) = w_i(t), & i \neq c \end{cases} \quad (1)$$

The $\eta(t)$ decreases with the number of iterations t , and its initial value and calculation method are consistent with those used in the unsupervised training phase. Following clustering, each cluster is assigned either a normal or attack label according to the majority class of the training samples contained within that cluster.

2.2. Job scheduling

PSTSA is applied to healthcare-cloud tasks that require computational resources, such as EHR retrieval, secure processing, encryption, and cloud-service execution. Requests classified as malicious by KSONN are discarded before entering the scheduling stage. Thus, only healthcare requests classified as normal by KSONN are forwarded to the subsequent modules of the MESBO-XACML framework for scheduling, access-control evaluation, and encryption. The XACML policy evaluation included test requests covering Permit, Deny, Not Applicable, and Indeterminate outcomes. Each test case is evaluated against the expected policy decision, and the observed decision is recorded for policy correctness analysis.

2.2.1. Prioritized Sorted Task Scheduling Algorithm

In this phase, job scheduling refers to prioritizing healthcare requests, including electronic health record retrieval, secure data access, encryption operations, and cloud service requests, according to user role and urgency. Moreover, the throughput values are reported in Mbps, and the graph axis is scaled accordingly to accurately signify the experimental measurements. The algorithm prioritizes healthcare tasks based on criticality and ranks them in decreasing order of execution durations. Requests classified as malicious by KSONN are immediately blocked, recorded in the security log, and excluded from scheduling, access-control evaluation, and encryption. According to Eqn. (2) the Expected Processing Time (EPT) matrix is equal to number of tasks multiplied by the number of VMs. Moreover, the EPT of a task change according to the computational capability of the allocated virtual machine and task execution requirements. The number of virtual machines VM is represented as $VM = VM_1, VM_2, VM_3 \dots VM_n$ and each virtual machine responsible for executing the scheduled healthcare tasks are represented in Eq. (2).

$$EPT(i, j) = \frac{Task_Length(i)}{Computing_Capacity_of_VM(j)} \quad (2)$$

For each healthcare task and virtual machine $VM(j)$, the expected processing time, waiting time, and expected completion time are calculated using the defined task and VM parameters. And then the waiting time of the current task on virtual machine j , i.e., $wt(j)$ is calculated using Eq. (3).

$$wt(j) = EPT(k, j) \quad (3)$$

Where k is index of job that comes before task i and is completed on same virtual machine; j and k denotes the index of the task previously come $EPT(K, j)$ denotes expected processing time of task k on virtual machine j . Eq. (4) provides the Expected Time to Complete (ETC) resulting from the distribution of task (i) over $VM(j)$.

$$ETC(j) = wt(j) + EPT(i, j) \quad (4)$$

The computational complexity of PSTSA is approximately $O(n \log n)$ because of task sorting, whereas task allocation proceeds in linear time according to the available virtual machines.

2.2.2. Fuzzy logic

After scheduling the job to the user, priority levels such as low, medium, and high are assigned using RFL. Multiple truth values can be processed through a single variable when fuzzy logic is used as a variable processing technique. Therefore, the fuzzy inference system considers the impact, urgency, importance, resource availability, critical patient condition, and deadline as input variables. Based on predefined IF-THEN rules, the system assigns low, medium, or high priorities to each healthcare request. Figure 3 illustrates the RFL architecture used for healthcare task prioritization and decision-making.

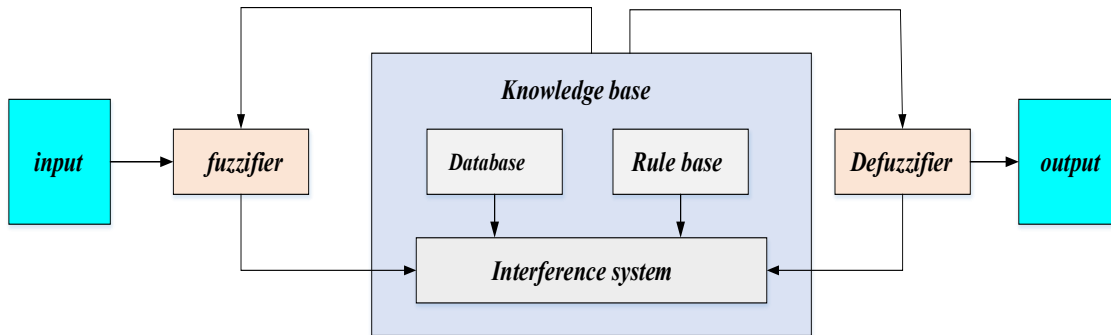


Figure 3. Architecture for Rule-Based Fuzzy Logic.

A mathematical model known as a fuzzy rough set, which produces a value between 0 and 1, is defined by the graded membership function. The throughput is calculated while processing healthcare records of diverse sizes under varying workloads to assess the scalability of the proposed framework. In contrast, a conventional set is typically described without graded membership, whereas in here Eq. (5) denotes X which is the total number of functions, μ_k represents the modified values for the fuzzy set.

$$X = \{(y, \mu_X(y)) \mid y \in X, \mu_X(y) \in [0,1]\} \quad (5)$$

where X denotes the fuzzy set, y denotes an element of the universe of discourse, and $\mu_X(y)$ denotes the membership degree of in fuzzy set, with values ranging from 0 to 1. For each healthcare request, the input variables are signified by normalized values in range [0,1]. For each. The resulting score is used to classify each request as low, medium, or high priority. These parameters are fuzzified using pre-established membership functions to identify the scheduling priority. The fuzzy inference engine processes the input variables according to the IF-THEN rules and produces a fuzzy aggregate, which is defuzzified by the centroid method into a crisp priority score. The priority score is then used to classify healthcare tasks into three categories: low, medium, and high. Consider the operation where X is X at the $X \in 1$, since operation X needs to be completed, the tasks that remain unassigned to X can be considered as part of set S is represented in the Eqn (6):

$$V_k = \min\{V_y + P_{ijz}\} \quad (6)$$

where V_k denotes the minimum accumulated scheduling value for operation k , denotes the current accumulated value, and P_{ijz} denotes the corresponding processing or priority contribution for the task assignment. Because operation X needs to be finished, the tasks that must be completed but are not assigned to X are considered to be in set S , the value of V_k is modified to update the set of remaining unassigned operations after each scheduling step, allowing the minimum number of operations among the remaining tasks to be selected for the next scheduling decision, and P_{ijz} is represented as the minimum number of operations X are illustrated in the Eqns (7)-(8):

$$S = \{X_{j1}, X_{j2}, X_{j3}, \dots, X_{ji}\} \quad (7)$$

$$V_k = V_k + P_{c*b,1} \quad (8)$$

Where s denotes set of remaining tasks and $X_{j1}, X_{j2}, X_{j3}, \dots, X_{ji}$ denote the remaining task elements. where V_k denotes the updated scheduling value and $P_{c*b,1}$ denotes the priority contribution obtained from the corresponding fuzzy-priority assignment.

V_k is modified to the value of S , the due date D_{pi} for jobs W_i can be determined using this fuzzy-priority model, and $P_{c*b,1}$ is represented as the values of the priority in the fuzzy set is represented in the Eqn (9):

$$D_{pi} = A_i + FQ_i \quad (9)$$

where D_{pi} denotes due date of job i , A_i denotes the arrival time of job i , and FQ_i denotes the fuzzy-priority-related scheduling contribution for job i . The fuzzy sets AI and Qi are fuzzified input variables processed by the fuzzy inference engine using

predefined IF–THEN rules. The inferred fuzzy outputs are aggregated and converted into crisp priority values using the centroid defuzzification method, enabling accurate healthcare task prioritization is illustrated in the Eqn (10):

$$D_{pi} = A_i + Q_i \quad (10)$$

Where D_{pi} denotes the due date of job i , A_i denotes its arrival time, and Q_i denotes the corresponding scheduling-time contribution. For healthcare task prioritization, the fuzzy inputs include task urgency, clinical impact, patient criticality, resource availability, and deadline proximity. The fuzzified input variables are processed by the fuzzy inference engine using predefined healthcare-specific IF–THEN rules. while Eqn. (11) can be used to map the setup times in matrix form.

$$Sut_i = \{SU_{xy}\} \quad (11)$$

The fuzzy input variable used to determine the priority of setup time is referred to as SU_{xy} . the amount of time needed to switch from work x to job y is SU_{xy} . Consequently, the setup time of the machine can be determined using Eqn. (12).

$$Sut_i = 0 \quad (12)$$

The output of the fuzzy-priority model ranges from 0 to 1 based on the evaluated input variables. The resulting priority scores are divided into three groups: low, medium, and high. The optimal work scheduling solution is obtained by using these priority scores to rank healthcare tasks. Based on the resulting priority levels, the rules used to assign task priorities are presented in Table 1.

Table 1. RFL rules for healthcare request priority measure.

Roles	Input						Output
	Impact (How much impact the role has on the system /task)	Urgency (How urgent is the task for this role)	Importance (How important is the job or action to be done?)	Availability (How easily available the person/resource is)	Critical cases (Are they handling critical/urgent cases?)	Deadline (How tight is the deadline for this role/task?)	Priority level
Admin	High	High	High	Low	High	High	High
Doctor	High	High	High	Medium	High	Low	High
Nurse	High	High	High	High	Medium	Medium	Medium
Managing staff	High	Medium	Medium	Low	High	Low	Medium
Training staff	High	High	Low	Low	Medium	Low	Low
Patient	Medium	High	High	High	High	High	High

According to the fuzzy rule assessment, healthcare service requests are assigned priority based on user's role. Administrative employees, medical officers, and patients receive high priority because their requests have a through impact on the arrangement of patient care services. Nurses and managerial personnel acquire medium priority level owing to the fact that their actions help carry out clinical practices but are not so urgent. Training specialists have a low-priority rating as they work only with educational or simulated data and do not treat patients.

2.3. Access control mechanism

Subsequently, XACML is developed to prevent unauthorized users from accessing medical and personal documents. System security can be increased while limiting unwanted user access to data using an ontology-based data access control model. The requested healthcare information is accessible only to authorized users; hence the XACML policy engine evaluates access requests from different healthcare users based on their designated roles and applicable access-control policies.

2.3.1. eXtensible Access Control Markup Language

this section explains that XACML is an XML-based language used to specify access-control policies and authorization rules for secure healthcare data access. Each access request contains attributes describing the subject, resource, action, and environment, along with their corresponding values. For example, the subject attribute may contain the user role such as doctor or nurse, the resource attribute may specify a medical record, the action attribute may specify view or modify, and the environment attribute may represent contextual conditions. The XACML policy engine is evaluated by implementing numerous healthcare access requests created by various user types, including doctors, nurses, patients, administrators, and trainers. During the policy evaluation process, the engine generates Permit or Deny decisions according to the predefined access-control rules. The rules of this policy are evaluated to determine whether a request fulfils its purpose; each applicable XACML rule produces one of four standard decision outcomes: permit, deny, not applicable, or indeterminate. Permit grants the requested access, deny rejects the request, not applicable indicates that no applicable rule determines the request, and indeterminate indicates that an error occurred during policy evaluation. Each rule consists of a target that defines the scope of the requests to which the rule applies and XACML estimates each access request by employing user attributes, resource attributes, environmental conditions, and ontology-based relationships before issuing permit or deny decisions through the PDP. If an error occurs during policy evaluation, the PDP returns an indeterminate decision because XACML is not designed for direct human authoring. Ontology-based policy generation is adopted to improve policy readability, consistency, and management. This suggests the need for automation to significantly reduce time and effort required for testing. Using a combinatorial method with values extracted from the policy, the automatic generation of requests suggests that ontology-based policy management can significantly reduce manual configuration errors while improving policy consistency. A brief explanation of the eXtensible Access Control Markup Language is given in Figure 4, where the architecture of eXtensible Access Control Markup Language is illustrated.

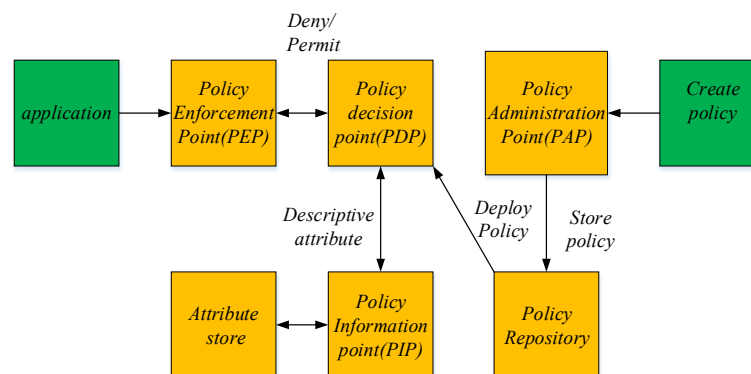


Figure 4. Architecture for Extensible Access Control Markup Language.

Hence, the XACML policy specifies the minimum and maximum quantities of identical entities that must be present in a request. For instance, if an XACML policy lacks any conditions or targets that require the evaluation of multiple subject entities, the minimum and maximum numbers of subject entities are two and three, respectively. Consequently, a robust method for deriving XACML test input requests is necessary, thus this approach can be utilized to test the PDP, which is the XACML policy execution engine, and to compare the responses of the PDP with the expected outcomes. However, creating a suitable set of XACML requests turned out to be a challenging task, and managing them manually is a significant amount of work.

$S_1 \rightarrow$ View & read his/her record

$S_2 \rightarrow$ Modify records

$S_3 \rightarrow$ Read & write prescription

$S_4 \rightarrow$ Add patient

Table 2. Access control rules for hospital management.

Role	View Personal record	View medical record	Read prescription	Write prescription	Modify medical record	Add patient
Hospital admin	✓	✓	X	✓	✓	✓
Doctor	✓	✓	X	X	✓	✓
Nurse	✓	✓	X	✓	X	X
Patient	✓	✓	X	X	X	X

Table 2 summarizes the access permissions associated with the healthcare roles in the proposed ontology-based XACML policy.

According to the ontology-based XACML policy, patients can view their medical records and change their authorized demographic details, including their address and contact information. However, patients cannot change the clinical diagnosis, treatment records, lab reports, prescriptions, or any medical information generated by doctors. This policy is responsible for maintaining data integrity while providing patients access to their healthcare information. The permissions include view medical records, read prescriptions, write prescriptions, update medical records, and add new patient records, according to the assigned user privileges.

2.4. Security Enhancement

In this paper, MESBO denotes the proposed Martino Encryption scheme optimized using SBO. SBO is used as the metaheuristic optimization algorithm for cryptographic key optimization, whereas Martino Encryption performs the encryption and decryption operations. Martino encryption with MESBO is designed to safeguard patient privacy and improve the security of data stored in cloud server. Then, MESBO algorithm is proposed that helps in generating safe encryption keys because of its capability to increase the randomness and entropy of keys along with increasing cryptographic strength. The fitness function maximizes the key entropy and randomness while minimizing the key correlation to generate highly secure encryption keys. However, quantifying these benefits is difficult because healthcare environments differ in terms of user requirements, security policies, and operational conditions.

2.4.1. Secretary Bird Optimization

SBO is a population-based metaheuristic optimization algorithm inspired by hunting and escaping behaviors of secretary birds. It consists of three phases: exploration, prey-searching, and exploitation. In the proposed framework, these phases are used to search for candidate cryptographic key representations according to the defined fitness function. Within the population-based metaheuristic approaches, the SBO algorithm consists of three optimization phases: exploration, prey searching, and exploitation. Let X_i denote current position of the i -th secretary bird and denote its updated position after applying corresponding optimization strategy. The objective-function value associated with candidate position is used to evaluate the quality of the candidate cryptographic key. In this study, optimization process incorporates hunting and escape strategies to explore and refine candidate key solutions. These two categories of actions include hunting strategies (exploration) and escape strategies (exploitation). The mathematical formulation for key generation using SBO is given below Equation (13).

3.4.1.1 Initialization

The population of the secretary bird is used to initialize the key value, and Eqn. (13) is used to mathematically formulate the initialization function.

$$K = \{K_1, K_2, \dots, K_n\} \quad (13)$$

where K denotes candidate cryptographic key, K_1 denotes the i -th key element, and $n = 128$ represents total number of bits in the cryptographic key. The key value is represented as K .

3.4.1.2 Fitness function

The fitness function for this optimization strategy is maximizing key entropy while minimizing the key correlation to create highly secure encryption keys are represented in the below Eqn (14).

$$\text{Information loss} = H(p) - H(q) \quad (14)$$

Where $H(p)$ denotes entropy of the plaintext distribution and $H(q)$ denotes the entropy associated with the transformed or candidate-key distribution. The fitness function is used to select the candidate key that provides high entropy and low information loss. The entropies of probability distributions are denoted as H_1 and H_2 , respectively. Therefore, the SBO algorithm utilizes a single fitness function to determine the optimal encryption key. The primary goal of the SBO algorithm is to maximize key randomness and entropy while minimizing correlation and repetition of key patterns, thereby enhancing cryptographic security. Consequently, the fitness function is defined as a combination of normalized entropy, key-correlation penalty, and key-repetition penalty, enabling the algorithm to determine secure encryption keys. The best candidate key is selected for encryption.

3.4.1.3 Position update

The position of each secretary bird is updated based on its fitness value using escape and hunting strategies.

3.4.1.4 Hunting strategy (Exploration)

Therefore, Eqn. (15) quantitatively simulated the position updates of the secretary bird during the attacking prey stage.

$$X_i = \begin{cases} X_i^{new,P1}, & \text{if } F_i^{new,P1} < F_i \\ X_i, & \text{else} \end{cases} \quad (15)$$

X_i is position of i^{th} secretary bird and F_i reflects value of the objective function that i^{th} secretary bird can attain. $X_i^{new,P1}$ symbolizes the i^{th} new state of the secretary bird, and the fitness value of the objective function is represented by $F_i^{new,P1}$.

3.4.1.5 Escape strategy (Exploitation)

The updated position represents the new candidate cryptographic key generated during the optimization process, and the position of each secretary bird is mathematically represented by Eqn. (16).

$$X_i = \begin{cases} X_i^{new,P2}, & \text{if } F_i^{new,P2} < F_i \\ X_i, & \text{else} \end{cases} \quad (16)$$

X_i is the position of the i^{th} secretary bird, and $X_i^{new,P2}$ stands for the current status of the i^{th} secretary bird. The objective function value obtained by the i -th secretary bird represents the fitness value of its candidate solution.

3.4.1.6 Termination

The optimization process terminates when the stopping criterion is satisfied or maximum number of iterations is reached. Figure 5 illustrates workflow of the SBO algorithm, including the initialization, fitness evaluation, exploration, exploitation, and selection of the optimal candidate cryptographic key.

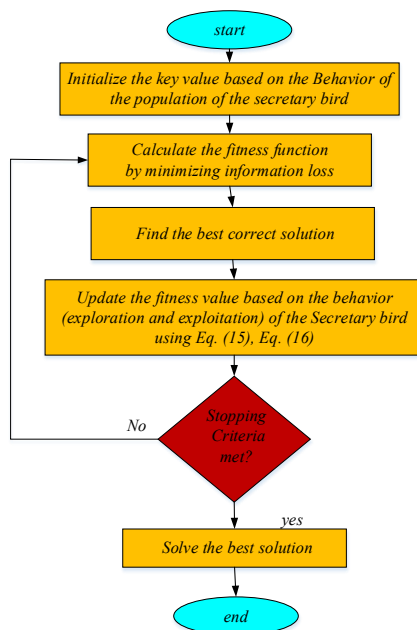


Figure 5. Flowchart for SBO.

2.4.2. Martino Encryption

The proposed MESBO-based Martino Encryption algorithm follows the Martino Encryption procedure described in [23], where binary conversion of ASCII values, key-dependent matrix transformations, rotation, transposition, and permutation operations are used for symmetric encryption and decryption. Mathematical models are used to jumble data so that only those with the necessary key can decode and access it, encryption is a data protection method. The cryptographic key is initialized using a secure random-generation procedure, whereas SBO is used only for the defined optimization objective associated with the encryption scheme. Moreover, a 128-bit secret key is generated by employing the MESBO algorithm and securely utilized by the Martino encryption scheme to provide resistance against brute-force attacks. Therefore, the encryption key is generated from a random initialization process and further optimized by MESBO before being utilized during encryption and decryption. The Martino encryption mechanism and its key-generation process using SBO are described below, and Figure 6 illustrates architecture of the Martino Encryption method.

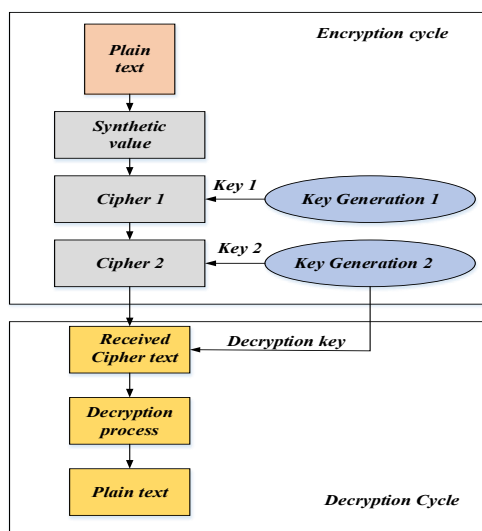


Figure 6. Architecture for Martino Encryption.

3.4.2.1 Encryption

Encryption converts readable plaintext into unreadable ciphertext to prevent unauthorized access. The input to the encryption process is plaintext or a secret message that is transformed into ciphertext using the proposed MESBO-based Martino encryption algorithm. The encryption configuration specifies the 128-bit secret key length, initialization vector/nonce generation procedure, block-processing mode, padding scheme, and authentication mechanism to ensure consistent and secure encryption and decryption. A unique nonce is generated for each encryption operation and is stored with the ciphertext but is not treated as a second. If the length of the binary number is less than eight, it is padded with zeroes to make it eight bits and it uses the modified binary for each character to create a " M " matrix.

The encryption procedure is performed as follows:

Step 1: Read plaintext healthcare data.

Step 2: Generate a 128-bit secret key using a Cryptographically Secure Pseudorandom Number Generator (CSPRNG).

Step 3: Convert the plaintext into binary representation.

Step 4: Apply key-dependent binary substitution, circular rotation, and permutation operations using the generated secret keys.

Step 5: Repeat the transformation process until all plaintext blocks are encrypted.

Step 6: Produce the ciphertext.

Step 7: The encrypted healthcare data are securely stored in the cloud for authorized access.

3.4.2.2 Decryption

Decryption restores the ciphertext to its original plaintext form using the corresponding secret key. Because the proposed cipher is symmetric in nature, same secret key is used during both encryption and decryption. First, ciphertext is separated into its encrypted blocks using the separator defined by the Martino encryption procedure. The binary digits obtained from the encrypted blocks are arranged into the matrix M_2 , whose dimensions are determined by the length of the binary sequence. The matrix is then transposed to obtain the intermediate matrix used for inverse transformation. Next, the inverse permutation and key-dependent rotation operations are applied according to the transformation sequence generated during encryption. A Rotate Shift Up (RSU) or Rotate Shift Down (RSD) operation is performed on M_2 according to the corresponding key digits and their parity. Finally, the resulting binary representation is converted back into character form to reconstruct the original plaintext.

Keep the final matrix in " $M1$ " and if ' ODD ' is third digit in the key, then RSU or RSD operation is proceeded on ' $M1$ ' as many times as the fourth digit in key. The final matrix is saved as " M ", it is then laid out and saved in the text field " T ." The pseudocode for the proposed MESBO-XACML framework, including the attribute-based access-control mechanism for secure healthcare data management, is presented below.

Algorithm1: Pseudocode for ontology-based attribute access control and MESBO encryption in healthcare data

```

Input: Data related to healthcare applications (H)
Start
#performing attack detection
{
AD= kohonen self-organizing neural network(H)
{
KS=Normal
NN=Attack
}
}
#performing a job scheduling
PS= Prioritized Sorted Task Scheduling Algorithm (KS)
{

```

```

RF=Rule-Based Fuzzy Logic (PS)
{
X=low
Y=medium
Z=high
}
}
#performing an access control mechanism
{
EA=eXtensible Access Control Markup language (XACML)
}
#performing a security enhancement
{
MS=MESBO-based Martino Encryption (EA) scheme optimized using Secretary Bird Optimization (SBO) #key
value based on the Secretary Bird Optimization
Initialize the  $K = \{k_1, k_2, k_3, \dots, k_n\}$  position based on the secretary bird
{
#Calculate the fitness function
Fitness =  $w_1 \times \text{Normalized Entropy} - w_2 \times \text{Key Correlation Penalty} - w_3 \times \text{Key Repetition Penalty}$ 
#Update value based on the behavior of the secretary bird
{
Value gets updated based on exploration behavior using Eqn. (15)
}
Else
{
Value gets updated based on exploitation behavior using Eqn. (16)
}
}
#termination
}
Output: Secure healthcare data are generated and stored after authorization and encryption.
End.
    
```

In the above algorithm, KS represents a set of legitimate healthcare requests accepted by the KSONN module, and *NN* indicates the malicious requests that are immediately blocked, logged, and excluded from further processing to maintain the security and integrity of the healthcare system. This algorithm iteratively optimizes the given fitness function until it finds the key with maximum entropy and minimum statistical correlation, improving the security of medical information.

3. RESULT AND DISCUSSION

The Martino encryption technique is employed by the MESBO in its cryptographic framework, which is crafted to improve the privacy of the patient and reinforce the security of data stored in the database. Python 3.8 is used to facilitate the testing, and the system specifications are RAM: 16 GB, GPU: NVIDIA GeForce GTX 1650, and CPU: Intel Core i5. The same train-test split is maintained for all comparative models to ensure a fair performance evaluation. All baseline models are trained utilizing identical preprocessing phases, the same train-test split, and similar experimental settings to confirm an unbiased performance comparison. For reproducibility, the experimental configuration should specify the dataset version, train-test split, random seeds, KSONN initialization parameters, ontology definition, XACML policies, fuzzy membership functions and rules, encryption configuration, and SBO parameters. To estimate robustness of proposed framework, the experimental results are reported as averages over ten independent experimental runs. Although KSONN is fundamentally a clustering-based model, cluster labels are mapped to attack and normal classes before evaluation, allowing a fair comparison with traditional models. Tables 3 and 4 present the simulation parameters of the KSONN and SBO algorithms, respectively.

Table 3. Simulation parameter for KSONN.

Parameters	Range
Input dim	11
Map size	[50 50]
Learning rate	0.02
epochs	50
Neighborhood radius	3

From the hyperparameter sensitivity analysis given in Table 3, it can be seen that a learning rate of 0.02 is chosen as the final model setting, as it shows consistently optimal performance in terms of validation. XACML policies are evaluated using predefined healthcare access-request test cases that cover authorized and unauthorized combinations of subjects, resources, actions, and environmental conditions. The expected and actual policy decisions are compared to verify the correctness of the policy. The evaluated learning rates are 0.02, 0.03, 0.04, and 0.05, with 0.02 selected for the final configuration. These parameter values are designated after the preliminary experiments to balance the attack detection accuracy and computational efficiency. A hyperparameter sensitivity analysis is conducted to determine the ideal training setup for the KSONN model. Fifty epochs are finally selected due to the fact that they achieved optimum validation accuracy without incurring excess computational expenses. The KSONN model utilizes 11 different input features: destination IP, Source IP, protocol type, packet size, flow duration, connection status, service type, packet rate, payload size, transmission interval, and traffic behavior.

Table 4. Simulation parameter for SBO.

Parameters	Range
Key Length	128 bits
Key representation	Binary vector of length 128
Number of populations	30
Number of iterations	100

Table 4 lists the simulation parameters of the SBO algorithm used for cryptographic key optimization. The cryptographic key is represented as a 128-bit binary vector, with each optimization variable corresponding to one key bit. Therefore, the optimization variable is a 128-dimensional binary vector, where each dimension represents one bit of the 128-bit cryptographic key. Therefore, the search space corresponds to the set of valid 128-bit binary key representations to deliver an adequate optimization search space for secure encryption key generation while sustaining computational efficiency. Parameters used in SBO are number of iterations, number of populations, upper limit, and lower limit.

3.1. Dataset description

The Cleveland Heart Disease Dataset [19] is used only for evaluating computational behavior of the proposed encryption and decryption operations. The dataset contains 303 records and represents structured healthcare information rather than actual hospital EHR documents. Therefore, the cryptographic experiments evaluate the computational behavior of the proposed encryption mechanism on structured healthcare records rather than validating confidentiality on real clinical documents. The encryption and decryption measurements are obtained using the processed Cleveland Heart Disease Dataset containing 303 records.

The dataset is used only for the stated decryption evaluation, number of records, and the corresponding record size. The class distribution is sustained during the train-test split to maintain class proportions and improve model generalization. Dummy values have been inserted in the place of the patient's name and social security number, which are removed from the database.

The Cleveland database is contained within a single processed file, although the directory also includes the original four raw files. In brief, each experiment is conducted on both datasets independently to measure the output of each module within the framework. The Receiver Operating Characteristic (ROC) curve of proposed KSONN attack-detection model on the Healthcare IoT Security Dataset is shown in Figure 7.

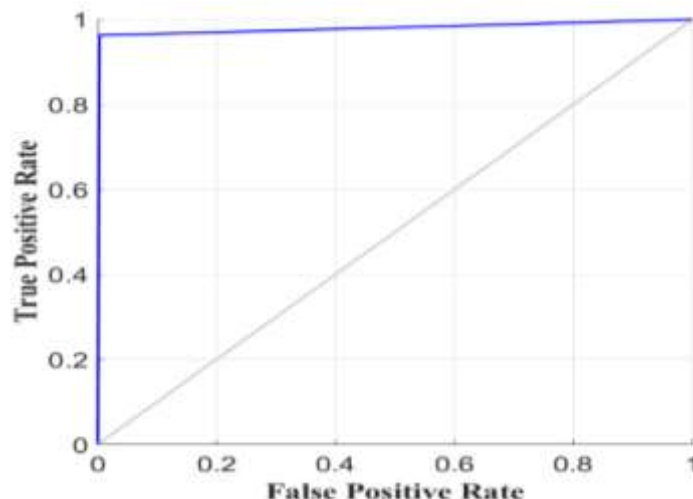


Figure 7. ROC curve of proposed KSONN attack detection approach on the Healthcare IoT Security Dataset.

Figure 7 illustrates the ROC curve, which evaluates classification performance of the proposed model across different decision thresholds. A high Area Under the Curve (AUC) value, approaching one signifies that the classification results of the model are highly accurate. The proposed KSONN model achieved an AUC of 0.98 on the evaluated test set, and the AUC confidence interval is calculated from the prediction scores of the model.



Figure 8. Confusion matrix of proposed KSONN

Figure 8 represents the confusion matrix of the KSONN attack-detection module, where attack traffic is considered the positive class. The confusion matrix contains 492 True Positives (TP), 409 True Negatives (TN), 3 False Positives (FP), and 16 False Negatives (FN), resulting in 920 test samples. Based on these values, the computed performance metrics are 97.93% accuracy, 99.39% precision, 96.85% sensitivity, and 99.27% specificity. The corresponding F1-score is 98.11%. The confusion matrix represents one representative testing run performed on 920 test samples, corresponding to 20% of the 4,600-sample Healthcare IoT Security Dataset. Table 3 presents the final simulation parameters used for the KSONN model. The hyperparameter sensitivity analysis is presented in Table 5. The learning rate of 0.02 is selected as the final model setting because it provided the highest observed performance among the evaluated learning rates.

Table 5. Values are reported as mean $\pm$ uncertainty over ten independent experimental runs.

Learning rate	Accuracy	Precision	Specificity	Sensitivity
0.02	97.93 $\pm$ 0.18	96.65 $\pm$ 0.21	97.24 $\pm$ 0.19	98.78 $\pm$ 0.16
0.03	96.76 $\pm$ 0.24	95.56 $\pm$ 0.27	94.87 $\pm$ 0.25	96.99 $\pm$ 0.23
0.04	95.88 $\pm$ 0.29	93.66 $\pm$ 0.31	92.45 $\pm$ 0.34	94.88 $\pm$ 0.28
0.05	94.11 $\pm$ 0.36	92.88 $\pm$ 0.38	91.65 $\pm$ 0.41	93.12 $\pm$ 0.35
Epoch	Accuracy	Precision	Specificity	Sensitivity
50	97.93 $\pm$ 0.18	96.65 $\pm$ 0.21	97.24 $\pm$ 0.19	98.78 $\pm$ 0.16
40	96.22 $\pm$ 0.26	95.76 $\pm$ 0.24	96.23 $\pm$ 0.22	97.77 $\pm$ 0.20
30	95.24 $\pm$ 0.31	94.89 $\pm$ 0.29	95.87 $\pm$ 0.27	96.99 $\pm$ 0.24
20	94.11 $\pm$ 0.37	93.67 $\pm$ 0.35	92.15 $\pm$ 0.39	94.29 $\pm$ 0.33

The values in Table 5 signifies mean performance across independent experimental runs, whereas the confusion matrix in Figure 8 represents one representative test run. Therefore, the metrics derived from Figure 8 should not be interpreted as the ten-run mean values. Table 5 presents the hyperparameter sensitivity analysis of proposed approach, where the learning rate is 0.02 and an accuracy of 97.93%, a precision of 96.65%, a specificity of 97.24%, and the sensitivity of 98.78%. As the learning rate increases, all the reported performance metrics decrease because excessively large learning rates reduce the convergence stability of the model. All values in Table 5 are reported as mean $\pm$ standard deviation over ten independent experimental runs. The manuscript specifies whether these values represent standard deviation, standard error, or confidence intervals. As the evaluated number of epochs decreases from 50 to 20, the reported performance metrics also decrease. Furthermore, the hyperparameter sensitivity analysis is conducted by changing the learning rate and training epochs to evaluate their influence on model performance. The access control framework is assessed through numerous user roles and healthcare access requests, representing its capability to efficiently process increasing numbers of authorization requests.

4.2 Comparative Analysis

To further verify effectiveness of proposed MESBO-XACML model, it is compared with the existing Context-Aware Risk Attribute Access Control (CARAC) [15] and Fog-Based ACAAC (FB-ACAAC) [16] models. The comparative performance of the proposed framework is obtainable in Table 6.

Table 6. Comparative Analysis of Proposed MESBO-XACML.

Model	Accuracy (%)	Precision (%)	Specificity (%)	Sensitivity (%)
Adapted CARAC [15]	93.84 $\pm$ 0.42	92.15 $\pm$ 0.47	93.06 $\pm$ 0.44	94.21 $\pm$ 0.39
Adapted FB-ACAAC [16]	95.67 $\pm$ 0.29	94.83 $\pm$ 0.31	95.42 $\pm$ 0.28	96.18 $\pm$ 0.25
Proposed MESBO-XACML	97.93 $\pm$ 0.18	96.65 $\pm$ 0.21	97.24 $\pm$ 0.19	98.78 $\pm$ 0.16

Table 6 demonstrates that the proposed MESBO-XACML model outperforms the adapted CARAC [15] and FB-ACAAC [16] models in accuracy, precision, specificity, and sensitivity. For comparative evaluation, CARAC and FB-ACAAC are

implemented as adapted baselines under the same experimental environment. The KSONN attack-detection component is applied consistently to all evaluated systems so that the reported accuracy, precision, specificity, and sensitivity refer specifically to attack-traffic classification. All comparative models used the same 80:20 train-test split, identical preprocessing, the same experimental configuration, and the same evaluation protocol. Ten independent experimental runs are conducted to ensure a fair and repeatable performance comparison.

4.2.1 Cryptographic Performance Evaluation

The proposed encryption scheme provides statistical transformation of the plaintext and ciphertext; however, formal cryptographic security requires additional cryptanalysis beyond the statistical evaluations presented in this study. The encryption performance is analyzed in terms of key entropy, ciphertext correlation coefficient, avalanche effect, encryption time, decryption time, and key-space complexity. These metrics collectively demonstrate the randomness, security strength, and computational efficiency of the proposed encryption mechanism to protect healthcare records. Table 7 presents the measured cryptographic performance of the proposed MESBO encryption scheme.

Table 7. Cryptographic Performance Analysis of the Proposed MESBO Encryption

Metric	Measured Value	Desired Interpretation
Byte-Level Shannon Entropy	7.9984 bits/byte	Very high randomness (ideal ≈ 8 bits for byte-wise data)
Correlation Coefficient	0.0018	Very low ciphertext correlation
Avalanche Effect (%)	49.87%	Close to the ideal 50%
Encryption Time (ms)	15.63 ms	Fast encryption
Decryption Time (ms)	14.98 ms	Fast decryption
Key Space	2^{128}	Provides a nominal 128-bit key space
Peak Memory Usage (MB)	23.6 MB	Low memory overhead

The reported encryption and decryption times should be interpreted together with the encrypted data size and measurement configuration, as these factors directly influence the computational time required by the encryption and decryption processes. The corresponding record size should be reported to support reproducibility and the encrypted output achieved a measured byte-level Shannon entropy of 7.9984 bits/byte. The ciphertext correlation coefficient of 0.0018 indicates very low linear correlation between the plaintext and ciphertext measurements, while the byte-level Shannon entropy of 7.9984 bits/byte indicates high measured randomness. The avalanche effect of 49.87% is very close to its perfect value being 50%, which means that the keys are highly sensitive. Moreover, the new encryption system has demonstrated the time of 15.63 ms for encryption and 14.98 ms for decryption with the peak process memory usage is 23.6 MB, respectively.

4.3 Statistical Analysis

A paired t-test is implemented for ensuring validity of performance of MESBO-XACML framework when compared to the existing context aware risk attribute access control model. Therefore, the computational cost of the proposed framework is analyzed at both module and end-to-end levels using execution time, memory consumption, and throughput measurements. When multiple access control policies conflict proposed framework follows a deny-overrides policy to ensure secure authorization decisions. The statistics related to the MESBO-XACML method are represented in Figure 9.

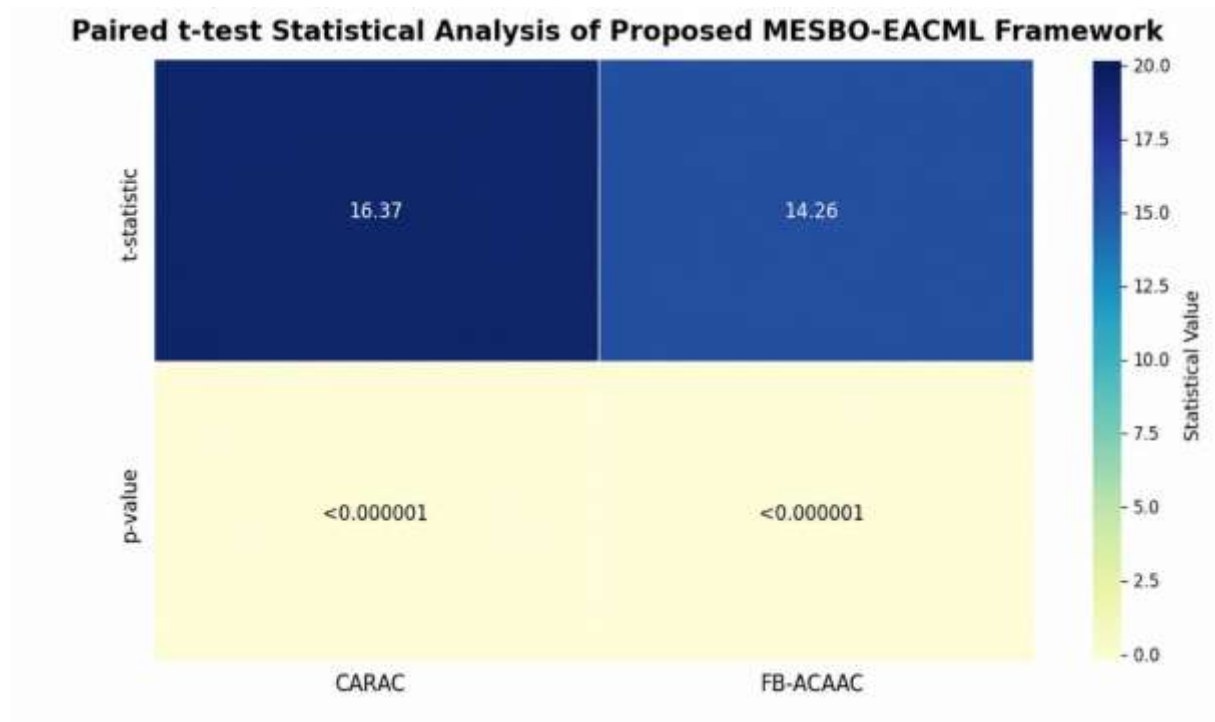


Figure 9. Paired t-test statistical analysis of the proposed MESBO-XACML framework.

The proposed framework is compared with adapted versions of CARAC and FB-ACAAC under a common KSONN-based attack-detection configuration. Moreover, ten independent experimental runs are considered for the paired t-test. Statistical significance is assessed using a paired t-test and the statistical analysis supports significant performance differences between the proposed framework and the comparison methods under the stated experimental conditions. All statistical comparisons are recalculated from the same ten paired experimental observations, and the resulting t-statistics, p-values, and confidence intervals are reported consistently in Figure 9 and Table 8.

Table 8. Statistical analysis of the proposed MESBO-XACML framework

Baseline	Mean Difference	Standard Deviation	t	Df	p-value	95% CI
CARAC [15]	3.21	0.62	16.37	9	<0.000001	2.77-3.65
FB-ACAAC	2.48	0.55	14.26	9	<0.000001	2.09-2.87

Table 8 presents the statistical comparison between the proposed MESBO-XACML framework and the CARAC and FB-ACAAC baselines using paired t-tests over ten independent experimental runs. Paired t-tests are performed using the ten corresponding run-level observations obtained under identical experimental conditions. The assumptions underlying the paired t-test are assessed before statistical comparison. The statistical results indicate significant differences between the evaluated methods under the stated experimental conditions. The mean performance differences are 3.21% points for CARAC and 2.48% points for FB-ACAAC. The

The mean performance differences are 3.21 % points for CARAC and 2.48 % points for FB-ACAAC. The corresponding t-statistics are 16.37 and 14.26, respectively, with nine degrees of freedom and p-values below 0.000001.

4.4 Discussion

This research proposes MESBO-XACML, an integrated healthcare-security framework that combines ontology-driven attribute-based access control, intelligent task scheduling, intrusion detection, and optimized cryptographic protection for

secure healthcare cloud environments. In particular, the KSONN provides effective attack detection by preserving the topological relationships among healthcare network traffic patterns. Classification metrics are used exclusively to evaluate the KSONN, while separate module-specific metrics are used for evaluating the scheduling, ontology reasoning, access control, and encryption performance. Ontology performance is evaluated using ontology size, reasoning time, consistency verification, and the correctness of inferred access-control relationships. Another innovative feature of the proposed framework is utilization of RFL, which delivers a dynamic method for prioritizing tasks, based on the urgency, significance, availability, and criticality of tasks assigned to healthcare workers. By using the XACML model. However, there are several challenges to consider relative to the practical implementation of these solutions in a large hospital infrastructure due to potential computational overhead, lack of compatibility across multiple electronic health record systems, and rapidly changing medical policies. Improvements to scalability and resilience may be achieved through future research into the evaluation of federated-aided, zero trust, and post-quantum encryption technologies. In a modern smart healthcare environment, the MESBO-XACML framework facilitates secure management of EHR, telemedicine services, diagnostics with cloud assistance, and patient monitoring through the IoT. The strong precision reported by the system shows that the false positive rate of detecting attacks in healthcare is low. Experimental results establish that the proposed incorporated framework effectively combines attack detection, intelligent task scheduling, ontology-based access control, and optimized encryption to deliver secure, reliable, and efficient healthcare data protection in cloud environments.

4. CONCLUSION

The proposed work provides a new ontology-based attribute-based access control model that is combined with intelligent attack detection, task scheduling with priority awareness, and cryptographic protection with optimization for secure management of health data. Subsequently, XACML is implemented to restrict unauthorized users from accessing sensitive information. Encryption and decryption times are measured for the evaluated Cleveland healthcare dataset to assess the computational efficiency of the proposed encryption scheme. To further safeguard patient privacy and improve data security within the database, a robust cryptographic technique is designed, utilizing MESBO and Martino encryption. The generation of the 128-bit cryptographic key is optimized using SBO, and the optimized key is subsequently used by the Martino Encryption scheme for encryption and decryption. The proposed encryption algorithm operates on a 128-bit binary key and the plaintext is converted into its binary representation and divided into fixed-size blocks. Each block is transformed using the key-dependent substitution, circular rotation, and permutation operations defined through proposed algorithm, the resulting transformed blocks constitute the ciphertext. Finally, the proposed framework attains a 97.93% of accuracy and 99.39% of precision in the representative attack-detection evaluation. Cryptographic performance is evaluated in terms of encryption time, decryption time, and memory usage. Experimental evaluation also measures the throughput and latency under the specified workload conditions. The proposed framework achieved 97.93% accuracy and 99.39% precision in the representative KSONN attack-detection evaluation. The proposed framework first detects malicious healthcare requests using KSONN, schedules legitimate tasks through PSTSA and assigns task priorities using RFL. Thereafter, it authorizes users through ontology-driven XACML access control, and finally protects healthcare data using the MESBO-based Martino encryption algorithm before secure cloud storage. The proposed framework has been validated in a simulated cloud environment. Future work will focus on deployment in practical hospital cloud infrastructures.

LIMITATION AND FUTURE SCOPE

The MESBO-XACML framework is a great solution with excellent security, performance, and reliability; however, there are still some limitations. First, there is a computational overhead associated with building multiple modules (KSONN, PSTSA, XACML, and MESBO) into the solution, which means they cannot easily be deployed in resource-constrained healthcare devices. The experimental results demonstrate the feasibility of the proposed framework in a simulated healthcare-cloud environment; validation in real hospital deployments remains part of the future work. The key-management process should define secure key generation, storage, distribution, rotation, revocation, recovery, and destruction procedures throughout the data lifecycle. Although the proposed framework attained better performance on benchmark datasets, its effectiveness in real-world hospital environments remains to be validated through practical deployment.

Future work will focus on deploying the framework in real hospital environments, extending ontology reasoning for dynamic access control, estimating larger healthcare datasets, and incorporating post-quantum cryptographic techniques to further strengthen security.

ACKNOWLEDGEMENT

Funding. The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Conflict of Interest. The authors declared that they have no conflicts of interest to this work. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Availability of data and material. Not applicable

Code availability. Not applicable

Author contributions. The corresponding author claims the major contribution of the paper including formulation, analysis, and editing. The co-author provides guidance to verify the analysis result and manuscript editing.

Compliance with ethical standards. This article is a completely original work of its authors; it has not been published before and will not be sent to other publications until the journal's editorial board decides not to accept it for publication.

REFERENCES

- [1] Torres-Charles, C.A., Sanchez-Gallegos, D.D., Gonzalez-Compean, J.L., Morales-Sandoval, M. and Carretero, J., 2025. Xook-Sec: A policy-as-code framework for secure data-sharing on the computing continuum: CA Torres-Charles et al. *Cluster Computing*, 28(14), p.889.
- [2] Renuka, S. and Guruprakash, C.D., 2024. Comparative Analysis of Ontology Based Data Access Control and Security Enhancement in Healthcare Application. *Optoelectronics, Instrumentation and Data Processing*, 60(2), pp.297-307.
- [3] Lim, K.S., Ooi, S.Y., Sayeed, M.S., Chew, Y.J. and Ahmad, N.M., 2025. Securing the Internet of Things: Systematic Insights into Architectures, Threats, and Defenses. *Electronics*, 14(20), p.3972.
- [4] Chiquito, A., Bodin, U. and Schelén, O., 2023. Attribute-based approaches for secure data sharing in industrial contexts. *IEEE Access*, 11, pp.10180-10195.
- [5] Xia, Z., Zhong, B., Zhao, T. and Zhang, S., 2026. Semantics-driven and identity masking attribute-based access control (SDIM-ABAC) in multi collaborative auditing (MCA) IIoT. *IEEE Internet of Things Journal*.
- [6] Tall, A.M. and Zou, C.C., 2023. A framework for attribute-based access control in processing big data with multiple sensitivities. *Applied Sciences*, 13(2), p.1183.
- [7] Zhang, R., Lu, C., Wu, Y. and Zhang, Z., 2025. An< Entity, Organization> Integrated Access Control Model. *Computers & Security*, p.104799.
- [8] Román-Martínez, I., Calvillo-Arbizu, J., Mayor-Gallego, V.J., Madinabeitia-Luque, G., Estepa-Alonso, A.J. and Estepa-Alonso, R.M., 2023. Blockchain-based service-oriented architecture for consent management, access control, and auditing. *IEEE Access*, 11, pp.12727-12741.
- [9] Ali, R., Ray, S. and Das, P., 2025, July. KPAAC CCN-IoT: Key Policy Attribute-Based Access Control Architecture for Internet of Things in Content Centric Network. In *International Conference on Computational Intelligence in Communications and Business Analytics* (pp. 17-31). Cham: Springer Nature Switzerland.
- [10] de Oliveira, M.T., Verginadis, Y., Reis, L.H., Psarra, E., Patiniotakis, I. and Olabarriaga, S.D., 2023. AC-ABAC: Attribute-based access control for electronic medical records during acute care. *Expert Systems with Applications*, 213, p.119271.
- [11] Mirani, U., Vora, N., Khanpara, P., Valiveti, S. and Raval, G., 2026. HDACS: A Secure and Efficient Framework for Attribute-based and Temporal Access Control in Healthcare Data Systems. *Telecommunication Systems*, 89(1), p.43.
- [12] Wang, F., Hu, Z., Wang, H., Chen, X. and Feng, W., 2023. Cross-domain dynamic access control based on “blockchain+ artificial intelligence”. *Neural Computing and Applications*, 35(35), pp.24575-24585.
- [13] Yang, Z., Chen, X., He, Y., Liu, L., Che, Y., Wang, X., Xiao, K. and Xu, G., 2024. An attribute-based access control scheme using blockchain technology for IoT data protection. *High-Confidence Computing*, 4(3), p.100199.
- [14] Zhang, X., Jing, C., Chen, Y.C., Wang, L., Xu, L. and Fu, D., 2024. Trusted Data Access Control Based on Logistics Business Collaboration Semantics. *Applied Sciences*, 14(10), p.4099.

- [15] Li, B., Yang, F. and Zhang, S., 2024. Context-aware risk attribute access control. *Mathematics*, 12(16), p.2541.
- [16] Kalaria, R., Kayes, A.S.M., Rahayu, W., Pardede, E. and Salehi Shahraki, A., 2024. Adaptive context-aware access control for IoT environments leveraging fog computing. *International Journal of Information Security*, 23(4), pp. 3089–3107. DOI: 10.1007/s10207-024-00866-4. . BGNBA-OCO based privacy preserving attribute-based access control with data duplication for secure storage in cloud. *Journal of Cloud Computing*, 13(1), p.8.
- [17] Kerl, M., Bodin, U. and Schelén, O., 2025. Privacy-preserving attribute-based access control using homomorphic encryption. *Cybersecurity*, 8, p. 5. DOI: 10.1186/s42400-024-00323-8
- [18] UCI Machine Learning Repository. Heart Disease Dataset, Cleveland subset, Dataset 45, 303 records.
- [19] Malik, F. IoT Healthcare Security Dataset. Kaggle
- [20] Ben-Shimol, L., Grolman, E., Elyashar, A., Maimon, I., Mimran, D., Brodt, O., Strassmann, M., Lehmann, H., Elovici, Y. and Shabtai, A., 2025. Observability and incident response in managed serverless environments using ontology-based log monitoring. *IEEE Transactions on Cloud Computing*.
- [21] Sanjalawe, Y., Al-E'mari, S., Abualhaj, M., Makhadmeh, S.N., Alsharaiah, M.A. and Hijazi, D.H., 2025. Recent advances in secretary bird optimization algorithm, its variants and applications. *Evolutionary Intelligence*, 18(3), p.65.
- [22] Rupa, C., Greeshmanth and Shah, M.A., 2023. Novel secure data protection scheme using Martino homomorphic encryption. *Journal of Cloud Computing*, 12, p.47. DOI: 10.1186/s13677-023-00425-7.