Journal of Information Systems and Telecommunication
Vol.12, No.2, April-June 2024, 90-104
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

FLHB-AC: Federated Learning History-Based Access Control Using Deep Neural Networks in Healthcare System

Nasibeh Mohammadi[1], Afshin Rezakhani[2*], Seyd Hamid Haj Seydjavadi[3], Parvaneh Asghari[4]

[1]. Department of Computer Engineering, Islamic Azad University, Boroujerd Branch, Borujerd, Iran
[2]. Department of Computer Engineering, Faculty of Engineering, Ayatollah Boroujerdi University, Boroujerd, Iran
[3]. Department of Computer engineering, Shahed University, Tehran, Iran
[4]. Department of Computer Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran

Received: 22 Oct 2023/ Revised: 04 Feb 2023/ Accepted: 03 Mar 2024

Abstract

Giving access permission based on histories of access is now one of the security needs in healthcare systems. However, current access control systems are unable to review all access histories online to provide access permission. As a result, this study first proposes a method to perform access control in healthcare systems in real time based on access histories and the decision of the suggested intelligent module. The data is used to train the intelligent module using the LSTM time series machine learning model. Medical data, on the other hand, cannot be obtained from separate systems and trained using different machine-learning models due to the sensitivity and privacy of medical records. As a result, the suggested solution employs the federated learning architecture, which remotely performs machine learning algorithms on healthcare systems and aggregates the knowledge gathered in the servers in the second phase. Based on the experiences of all healthcare systems, the servers communicate the learning aggregation back to the systems to control access to resources. The experimental results reveal that the accuracy of history-based access control in local healthcare systems before the application of the suggested method is lower than the accuracy of the access control in these systems after aggregating training with federated learning architecture.

Keywords: Healthcare System; History-Based Access Control; Intelligent Module; Deep Recurrent Networks; Federated Learning.

1- Introduction

A health information system (HIS) is a data management system for healthcare. This comprises systems for collecting, storing, managing, and transmitting a patient's electronic medical record (EMR); hospital operational management systems; and systems that support healthcare policy choices. Health information systems also include data management systems for healthcare practitioners and organizations. These technologies, for example, might be used to enhance patient outcomes, inform research, and impact policy and decision-making. Security is a critical concern in health information systems because they typically access, analyze, or store a substantial amount of sensitive data [1]. Access control refers to a set of procedures used to determine who or what has access to, uses, or changes what resources [2]. Access control is a critical topic in the design debates of physical security, information security, and network security to limit risks and threats. With the rapid advancement of computing and information technologies, classic access control models have become insufficient in terms of severe security needs, and new applications. ABAC models provide a more flexible way to deal with the authorization requirements of complex and dynamic systems. Modern access control systems and feature-based systems are becoming more popular [3,4]. Organizations are interested in adopting systems to regulate access to their resources that can best meet these demands as the use of access control systems expands in many industries such as IoT, cloud computing, and health care systems. During the COVID-19 time, for example, many businesses need a flexible approach to accessing software resources,

✉ **Afshin Rezakhani**
rezakhani@abru.ac.ir

allowing the user access to be adjusted based on the circumstances [5]. One of the existing challenges in access control systems is to use the previous accesses to grant the access permission at the next stage online and without interruption. Also, protecting patients' privacy while checking previous accesses is another challenge facing this article, which we try to solve it. The contributions of this paper are as follows:

- ✓ Using access histories in granting access permission to healthcare information systems.
- ✓ Using the deep recurrent network and intelligent module to provide online/real-time access requests.
- ✓ Maintaining the privacy of patient data in the HIS system and using local patient data to train the learning model
- ✓ Using federated learning to consolidate the training of local healthcare systems and improve the accuracy of the access control

The rest of this paper is as follows: Section 2 presents the background and fundamental concepts used in the paper background. In section 3, the surveys of related works are considered. The suggested approach and its components are explained in section 4. In section 5, the performance results and comparison with previous works are described. A discussion of our work is presented in section 6 and finally, section 7 concludes the paper.

2- Background

The basic ideas utilized in the article are defined in this section. The definitions of ABAC access control systems are explored first, followed by deep recurrent neural networks and federated learning model.

2-1- Attribute-based Access Control

The attribute-based access control (ABAC) is a type of access management model in which permission to perform a set of operations is determined by analyzing the attributes assigned to requesters, resources, and requested activities, as well as environmental conditions in some cases. Attributes are qualities that indicate a specific aspect of the requester and objects, environmental circumstances, or requested actions that the administrator has already established and assigned [4][6].

- ***Definition 1. Entities and Attributes:*** U, O, C, and OP are the system users (requesters), requested resources, defined particular conditions, and requested operations in the ABAC access control paradigm. Also, A_u, A_o, A_c, and A_{op} are the attributes of the requester, source, condition, and requested operation, respectively. Also, $E = U \cup O$, $\cup$ is the set of all entities, and $A = A_u \cup A_o \cup A_c \cup A_{op}$ is the set of all attributes of the entities mentioned above.
- ***Definition 2. Mapping Function:*** If $a \in A$ and $e \in E$, $F_{e,a}$ is the mapping function of the entity e on the attribute a. More precisely, the function $F_{e,a}(e,a)$ returns all the items that are related to the existence of e on attribute a. For example, $F_{a,e}$(Ali, location) = Tehran means that the location related to Ali is Tehran.
- ***Definition 3. Access request(req):*** An access request is a <u, o, c, op> where a requester u has requested operation op to access resource o in condition c. For example, <Ali, db1, Tehran, read> is an access request by Ali, who wants to read access to the source db1 from Tehran.
- ***Definition 4. Conventional access policy:*** Access permission is a sample(sample) sample=<req, g> decision g on request req_i in the access control system. sample can also be considered as access history and shows the access details.
- ***Definition 5. History-based access policy:*** If the access request is based on the records of sample <u, o, c, op>, the user u has requested the operation op to access the source o in the condition c where there is a c_j in c and it is essential to check the histories or check the prior accesses. For instance, *<db1, location.NewYork, $count_{100}$(access(db1, location(Tehran))<2), read, permit>* is an access policy of this type. If two conditions are met, the requester is granted access to db1. First, the requester's location must be in New York, and the number of accesses on db1 from the Tehran location should be no more than twice in the preceding 100 accesses. As a result, the access policy is established as <u, o, c, op, g> granting the requester g access in exchange for op access on o.
- ***Definition 6. Policy Repository(pr):*** A database containing all of the policies described in definitions 5 and 6. This repository's policies are organized into two broad groups. The first category includes only attributes related to the requester, resource, condition, and the requested operation (as seen in definition 1, we have designated the set of all these attributes with A), and the second category includes attributes that require checking access histories, which we denote by LA.
- ***Definition 7. History Access:*** If the current access request was made at time t, the access history provides a list of all accesses made from time t-k to time t-1. For example, if R_{t-k} access is provided at time t-k and R_{t-1} access is granted at time t-1,

the following access histories are defined at time t: $Hi(t)=\{R_{t-k}, R_{t-k+1}, \ldots, R_{t-1}\}$

2-2- Recurrent Neural Networks

Recurrent neural networks (RNNs) are artificial networks that are utilized in speech recognition, natural language processing, and sequential processing [7]. However, RNN features a feedback layer that feeds back the network's output as well as the next input. RNN can recall its previous input and use this information to process a sequence of inputs because of its internal memory. Simply said, RNNs incorporate a feedback loop that ensures that past knowledge is not lost and remains in the network. The following are the architecture types of RNNs.

2-2-1-Simple Recurrent Neural Network (RNN)

This is the most basic sort of recurrent network, yet it is still a viable alternative due to its modest number of parameters (when compared to GRU and LSTM networks) and reasonable accuracy in simple situations and short time series. The fundamental issue with simple RNNs is their limited memory, which results in vanishing and exploding gradients [7].

2-2-2-Long Short-Term Memory (LSTM)

LSTM networks are an upgraded version of RNNs that improves memory recall. In this sort of network, the problem of progressive fading of RNNs has been solved. LSTM is appropriate for time series categorization, processing, and prediction in the presence of time delays of unknown duration [8]. In addition, the LSTM network cell's inputs and outputs are as follows.

2-2-3-Gated Recurrent Units (GRUs)

The GRU recurrent neural network, like the LSTM, is intended to alleviate the RNN's short memory problem. Hidden layers are employed to handle and categorize input in the gated recurrent network rather than the state cell [9][10].

2-3- Federated Learning

The purpose of federated learning is to train a machine learning algorithm, i.e. deep neural networks, on multiple local datasets that exist at local nodes without explicitly exchanging data. The general principle consists of training local models on local data samples and exchanging parameters (e.g. weights and biases of the deep neural network) between these local nodes at some frequency to produce a global model shared between all nodes. The main difference between federated learning and distributed learning is in the assumptions made about the properties of the local datasets because the main goal of distributed learning is to parallelize the computing power whereas federated learning initiative aims to train heterogeneous datasets. While the goal of distributed learning is to train a single model on multiple servers, a common underlying assumption is that the local datasets are identically distributed and have approximately the same size. None of these hypotheses were made for federated learning. Instead, data sets are typically heterogeneous and their size may span several orders of magnitude. In addition, clients involved in federated training may be unreliable as they are subject to more failures or dropouts compared to distributed learning where nodes are typically data centers with powerful computing capabilities and are connected to fast networks [11] [12].

2-4- Healthcare System

A Healthcare platform enables doctors and their assistants to analyze data. Such an infrastructure should have things like simple equipment management, simple connections, data analysis, and intelligent data transformation [13]. Due to the vast amount of information, healthcare platforms must have the ability to accurately and timely analysis to provide the best analysis of various conditions. An overview of IoT-based healthcare is shown in Figure 1. The following components are essential in healthcare systems:

- ✓ Data collection using existing sensors
- ✓ Supporting a simple user interface for use by all patients and medical centers
- ✓ Access to infrastructure services and network services for all nodes in the network
- ✓ Increasing reliability, accuracy, durability, and strength in data storage and transmission.

Figure 1. An overview of a typical IoT-based healthcare system [14]

2-5- Related Works

This section will look at some recent studies in the field of the present subject. The authors of [15]

presented a blockchain-based access control system for GWAS with BFGF federated learning in their paper. Before training the local models, the framework uses automatic quality control (AQC) to assure the quality of the training data in this technique. It creates a blockchain authentication system to filter people. The authors of [16] focused on data security and privacy in industrial IoT systems utilizing machine learning models and federated learning models. They also investigated and contrasted other ways. A novel middleware for risk-based authorization and federated learning for health care (FRAMH) has been presented in [17], which provides risk-based access control for medical records. The authors employed a federated learning approach to assess health status risk and integrated it with blockchain to prevent unwanted access. Another study [18] offered "Hash and Signature-Based Policy-Based Encryption (hCP-ABES)," a cloud-based healthcare system for secure data storage and access control. The authors' proposed access control provides users with security, authentication, and secrecy when using medical data. Encryption and auditing procedures are employed to maintain the confidentiality and integrity of stored information. Access control mechanisms are usually used to govern data access during the data-sharing phase. In the data analysis process, machine learning algorithms are used to secure the privacy of massive medical data [19]. The authors of [20] suggested a framework to address issues such as information leaking in access granting. The authors suggested a federated learning framework for access control policies as well as a formal explanation of the policy transfer problem in attribute-based access control. Recent breakthroughs in the field of federated learning for cyber security and IoT security have been thoroughly addressed in [21]. This study's primary focus is on security, but it also explores different techniques for addressing FL-related performance difficulties that may compromise IoT security and performance. Another area of study is federated learning use in industrial systems [22]. This research looks into the FL prospects for next-generation networked industrial systems, as well as the problems of collaborative driving in connected and robotic autonomous vehicles. An approach for leveraging federated learning as a service with Decentralized Identities is proposed in [23]. The authors presented a DID-eFed system in which decentralized identities (DID) and a smart contract facilitate FL. DID offers flexible decentralized access management in the proposed system, and the smart contract provides a process with a few errors. [24] investigates FL in-depth, focusing on applications and operating systems, methods, and real-world applications. FL generates robust classifiers without requiring information disclosure, resulting in extremely secure privacy policies and access control rights. The authors in [25] propose a novel approach of combining CNN-LSTM with particle swarm optimization in the RBAC system. The convolutional neural network has extracted parsed SQL queries and long short-term memory was also suitable for modeling the temporal information of SQL queries. The paper [26] has considered an access control model for multi-channel heterogeneous networks based on deep reinforcement learning, referred to as multi-channel deep-reinforcement learning. To overcome the challenges of securing IoT devices, the authors in [27] have suggested a deep learning–based intrusion detection system to detect security vulnerabilities in IoT. The research [28] has protected the agreements dependent on ERC20 of controlled Ethereum-based Distributed Ledger Technology with cycles and capacities to get an all-surrounding framework for creating sure Cloud-Based Manufacturing jobs. effective attribute-based encryption is suggested in [29] which places part of the cryptography in the edge nodes as well as supports attribute updates and flexible control. To address the problem of scalability in access control, the authors in [30] have proposed an enhanced Bell–LaPadula model and categorized the peers and transactions in different clearance and security levels. In the article [31], the authors have proposed a blockchain-based approach that provides a decentralized EHR and smart-contract-based service automation without compromising the system's security and privacy. The paper [32] discussed some gaps within the existing access control strategies for health care. To fill this gap, the authors have proposed a secure access control model to control access in the healthcare system. Their solution has used the location of the user for providing secure access control views in the healthcare system. The article [33] has planned to create a novel solution based on blockchain technology that locates the patient in charge of granting and revoking access permissions for healthcare enterprises and providers to meet privacy regulations. Motivated by the research gaps, the paper [34] proposed a scheme, that integrates a blockchain (BC)-based confidentiality-privacy (CP) preserving scheme. In the paper [35], the authors have discussed how leveraging blockchain for healthcare data control can lead to better improvements. they presented the key blockchain features and characteristics. The paper [36] has focused on privacy issues in smart context-aware healthcare within the Electronic Transfer of Prescriptions. The access control models may expose user privacy to an attacker. To tackle this problem,

the authors [37] have used the cuckoo filter to disguise the right of entry policy to safeguard the private information of the owner. The inference assault has affected medical records. The paper [38] has proposed a new blockchain-based lightweight access control model. The scheme has used blockchain to create a trusted network by a special mechanism. The paper [39] reviewed the recent trends and critical requirements for blockchain-based and IoT access management. The authors showed several important views of blockchain, including decentralized control, secure storage, and sharing information for IoT access control. Deep learning and artificial intelligence frameworks are introduced in the paper [40] to improve cyber security such as access control. The authors in [41] have proposed a novel model by implementing a specific cryptography algorithm in which they used the Key generation scheme of RSA to encrypt health data. The paper [42] has proposed a deep learning-based anomaly detection method composed of estimation and classification models applied to a subdomain in healthcare systems. the authors [43] have conducted a universal review of federated learning systems. To get a clear flow and guide future probes, they introduced the definition of federated learning systems and analyzed the system sections. The authors of [44] presented a method for automatically learning ABAC policy rules from the access logs of a system. In the proposed approach, an unsupervised learning-based algorithm is used to recognize patterns in extracting ABAC authorization rules. In [45], an efficient and simple method has been proposed to verify the access control policy using a machine learning classification algorithm. Cotrini et al. [46] suggested an approach for deriving some rules from randomly distributed histories. In [49], a novel method was suggested for secured and integrated access control in the SIEM. The key points where the SIEM accesses the information within the software were specified and policies for access control were developed. To achieve energy efficiency in the network some simplification strategies have to be carried out not only in the Medium Access Control (MAC) layer but also in the network and transport layers [50].

3- Problem Definition

Although giving access authorization in HIS systems necessitates the use of feature-based access control, only limited features such as the requester's location and time have been used to validate access permission thus far. In the first step of this research, "access history" is proposed as an essential component in HIS system access control. A pulse based on a log, for example, is described as "a certain drug is administered to a patient if it is prescribed by at least two doctors." For such a pulse, the prior logs must be examined. To study access control policies based on log conditions, LSTM and GRU deep learning models are utilized, which must be triggered live during access requests. Deep learning is employed since you cannot spend a lot of time verifying access histories while seeking access. As a result, memory neural networks such as LSTM and GRU will be useful.

The limited number of samples available to train the learning model in local HIS systems presented a challenge during the initial step. Furthermore, because of privacy concerns, it is not viable to gather all medical data in HIS systems and train the intelligent learning model (previous step). As a result, the federated learning architecture is used in the second stage of the proposed approach, and instead of transferring data to the server, the suggested model is produced in the server and delivered to the HIS systems. After training the model in HIS systems, the model's output is forwarded to the server, which aggregates the results.

3-1- Challenges and Requirements

To develop log-based access control that can be applied to a wide range of real-world scenarios, we identify the following challenges and requirements:

- Online access request: Log-based resource access requests are submitted online and require a prompt, no-delay response with a short time order. However, the problem of the access control system based on access histories necessitates a review of past logs as well as the processing and time loads. As a result, the proposed model should be able to handle a huge number of online queries.
- Correctness in log-based access control: Because the extracted access decision must result in the same access decision as the policy repository, the suggested log-based access control system's prediction (response) must be compatible with the result of the original permission in the policies. An inconsistent answer may arise in instances where previously approved access is refused (more restrictive) or the system allows unlawful access (less restrictive).
- The complexity of log-based policies: The complexity of log-based policies is one of the most important challenges in access control systems. In many policies, it is not required to check logs, but in some policies, logs should be

checked. Such conditions will cause complexity in meeting all the conditions with a reciprocal effect on each other.

- Privacy Violation: Considering that medical records in HIS systems contain private information of patients, it is important to protect privacy during research and data review.

3-2- Evaluation Metrics

How well these accesses match the original accesses is one of the metrics for evaluating the quality of the accesses provided in the proposed approach (given by the original pulses). In other words, the proposed method's access result is compared to the original pulses' access result, and the quality of the suggested access control system is evaluated. For example, if the suggested method's prediction for an access request is to grant permission and access permission is granted in the main policy, the quality of the proposed approach will improve. The following definitions are taken into account for a more complete study of the proposed approach's evaluation metrics.

- ***Definition 8 (TP definition):*** If an access request is req_i=< $attr_s$, $attr_o$, opr, act > and the proposed approach prediction for req_i is equal to $pred_i$, and also if the original access policy is rul_j=< $attr_s$, $attr_o$, opr, act> and the label (access) of the original policy equivalent to the request is equal to lab_j, then TP is defined as follows.
  $TP_{<req,pred>}$ = | <r> ϵ DS | (lab(rul(r)) == permit) && (pred(req(r)) == permit) |
- ***Definition 9 (Definition of TN):*** Despite the assumptions of Definition 8, TN is defined as follows. This means that the prediction made by the proposed approach with the original policy label corresponding to that request is equal to *denial*.
  $TN_{<req,\ pred>}$ = | <r> ϵ DS | (lab(rul(r)) == deny) && (pred(req(r)) == deny) |
- ***Definition 10 (Definition of FP):*** Despite the above assumptions, FP is defined as follows. This means that the prediction made by the proposed approach is *permitted*, but the label of that request in the original policy is equal to deny, and it shows the wrong prediction in the proposed approach.
  $FP_{<req,\ pred>}$ = | <r> ϵ DS | (lab(rul(r)) == deny) && (pred(req(r)) == permit) |
- ***Definition 11 (Definition of FN):*** If the prediction made by the proposed approach is *denied* and the label of that request in the original policy is equal to permit, it means a wrong prediction in the proposed approach, which is defined as follows.
  $FN_{<req,\ pred>}$ = | <r> ϵ DS | (lab(rul(r)) == permit) && (pred(req(r)) == deny) |
- ***Definition 12:*** by calculating the above metrics, Accuracy = $\frac{TP + TN}{TP + FN + TN + FP}$ can be defined to determine the accuracy of the proposed approach.

Table 1 presents the symbols used in this article.

Table 1. Notations

Notation	***Definition***
U, O, C, OP	Users(requesters), resources, conditions, and operations
A_u, A_o, A_c, A_{op}	attributes of the requester, source, condition, operation
E, A	*U* ∪ *O*, set of all attributes of the entities
a, *e*, c_j	a∈A, e∈E, c_i∈c
G	Access permission
$F_{e,a}$	Mapping function of the entity e on the attribute a
C	Conditions in access histories
t, k	Time
R_i	Access in time i
Hi	access histories in time t
m_i	Equivalent method with any condition C
M	set of methods {m1, m2, ...}
F#, S#	The number of data features, The number of data samples
req_i	Access request i
Q_i	Set of features reqi
Train_Set	Training dataset in conditional dataset
NLA	Set {nla1, nla2, …}
LA	Set {la1, la2, …}
pr	Policy Repository
PAP	Policy administration point
PEP	Policy enforcement point
PDP	Policy decision point
PIP	Policy information point
DT, KNN, NN, SVM	decision tree, k-nearest neighbor, neural network, Support vector machine
$a^{<t>}$, $c^{<t>}$	Current stage status
$a^{<t-1>}$, $c^{<t-1>}$	The status of the previous stage
$x^{<t>}$,$\hat{y}^{<t>}$	Input, output (prediction) of the recurrent network
w_a, W_o, W_f, W_u, W_y, w_c	Weight vectors
$_t\hat{c}$	status of the new stage
σ	sigmoeid activity function
$_{u\Gamma}$, $_{f\Gamma}$	Update gate, Forget Gate
b_y, b_u,b_c, b_a	Bias
g_1,g_2	tanh activity functions

4- Intelligent FLACH Approach

This study is primarily concerned with providing access permission in healthcare systems based on the aggregation of local healthcare systems' knowledge and experiences. Because medical records are sensitive and private, it is not practical to collect data from various healthcare systems and train them using machine learning models. The federated learning approach is proposed in this paper to employ machine learning algorithms remotely on distinct local systems and aggregate the knowledge gained in

the servers. The servers re-send the aggregation of various learnings to the clients to regulate resource access based on the experiences of all local healthcare systems. Figure 2 depicts the general architecture of the suggested approach, which we call FLACH.

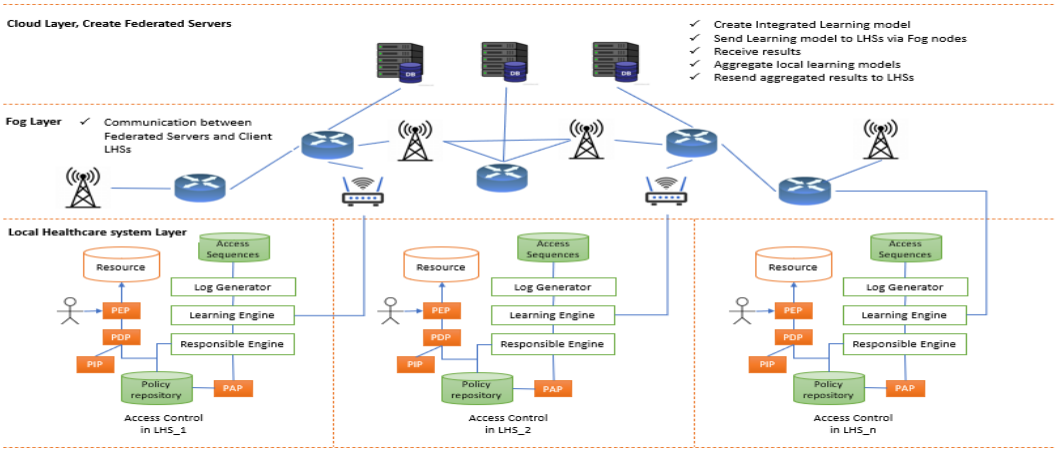

Figure2. Proposed multi-layer architecture

Before delving into the FL architecture, some key assumptions must be addressed. 1) Using access histories to authorize access is one of the requirements of healthcare systems. A nurse, for example, can inject a specific drug into a patient if it has been ordered by two general practitioners or a specialist. 2) Deep learning models based on time series, such as Simple RNN, LSTM, and GRU, are appropriate for granting access based on previous access histories. 3) Learning models like LSTM and GRU employ the order of subsequent accesses to grant access authorization. 4) The privacy of medical data is a distinguishing aspect of healthcare systems. As a result, patient information cannot be withdrawn from the local healthcare system.

The following sections describe the various components of the suggested method (which are explained in depth in the next section)

1. Federated learning architecture for LHS knowledge aggregation: The data is trained locally in each LHS's access control system before being utilized to give access. However, the difficulty with this proposal is the lack of limited and sensitive data, resulting in inadequate training for the proposed intelligent module. As a result, the technique of federated learning architecture is presented to aggregate the knowledge of intelligent access control systems (which exist in local healthcare systems).
2. An intelligent module within the access control system: to train the access control system, deep learning methods based on time series, such as LSTM or GRU, are used. When the policy repository incorporates access control rules based on checking access histories, the intelligent module in the access control system is required and unavoidable. As a result, an intelligent module for each local healthcare system is installed alongside the access control system.
3. Customized access control: When intelligent modules are installed in local access control systems and the knowledge of various modules is combined, the intelligent access control system is customized.

4-1- Design of Federated Learning Layers

First, servers are created in the cloud layer to aggregate training in local healthcare systems. These servers are cloud-based and have no access to medical records (data) in healthcare systems. These servers' primary function is to build a machine learning model based on time series and then distribute it to all local systems.

First, using Algorithm 1, a machine learning model based on time series is generated in the cloud layer servers. This article focuses on the use of LSTM and GRU models. In this algorithm, the learning model is defined first, followed by the necessary pre-processing. The necessary layers are then inserted, and the model compiles and begins training.

Algoithm1. Machine learning model created in server in the smart machine

Procedure *Create-Model ()*
Input: *Conditions, dataset*
Output:
Forever ()
{
model ← Sequential ()
dataset ← Preprocessing(dataset)
model ← CreateModel ()
model. Add (layers, optimizers, activations, …)
…
model.add (Dense, Dropout, …)
model. Compile (Train_Set)
model. Fit (Train_Set)
}

The model developed on the server is transferred via fog layer nodes to the intelligent module of local healthcare systems. There is no data transfer in this transfer; just the learning model is sent to the fog nodes to be sent to the local LHS systems .

After training the models in the intelligent module of the local systems, the results are transmitted back to the servers to be aggregated and re-distributed to the local systems via the fog layer nodes. Figure 3 depicts the sequence of generating the model on the server, delivering it to the LHSs, and resending the trained weights from the LHSs to the servers via the fog layer nodes.

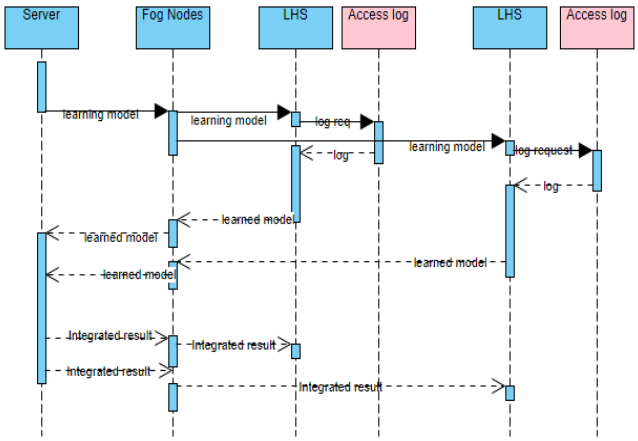

Figure 3. The sequence number of training models transfers from the server to the local healthcare systems

4-2- Intelligent Module Design

As previously stated, client layer design is a component of federated learning architecture. The proposed intelligent module is the federated architecture's client component. As a result, at this stage, the intelligent module is generated locally within the healthcare system's access control system. The learning engine, reaction engine, and access logs are all part of the local smart module. Machine learning models based on time series are assembled and run in the learning engine. It trains the learning model in the engine using local access records. In addition, the response engine will be employed in the access control system to answer access requests. When the policy repository contains policies that need access histories to be reviewed, the response engine is used (rather than checking all previous accesses). Figure 4 depicts the proposed module's components and their connections.

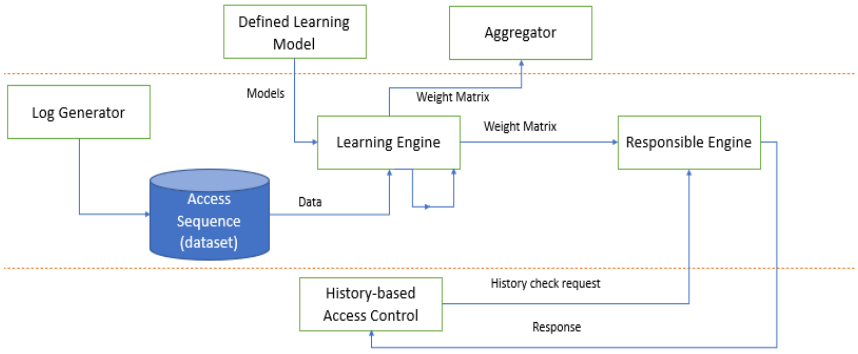

Figure 4. Components and connections of the intelligent module within the access control system

4-3- History-based Access Control Design

The proposed access control system uses an attribute-based model (access history). According to this model, the requestor sends the request to the policy enforcement point or PEP. This component sends the request to the PDP. PDP makes decisions through two main units policy repository and PIP. The PIP informs the current status of the requester and the resource. Also, all policies are in the policy repository.

So initially, the PIP delivers all the requester and resource attributes to the PDP. The PDP then retrieves access control policies associated with the requester from the policy repository. If the history corresponding to the requester in the policy repository does not contain log-based properties, the access permission is granted directly from the repository and there is no need to call the intelligent module's response engine. However, if the requester's history in the repository contains log-based properties, the PDP refers to the response engine, and the response of this engine is issued as a result of the access permission. Algorithm 2 shows the implementation of the access control system, which can implement both the current ABAC systems and the systems based on access histories.

Algoithm2. Access control algorithm

Procedure *AccessControl()*
Input: $request_i$
Output: *grant*
    $req_i \leftarrow request_i$
    PEP sends req_i *to PDP*
    PDP asks information of req_i *from PIP*
    $Q_i \leftarrow PIP\ (req_i)$
    For all $\boldsymbol{pr}_k \in \boldsymbol{policyRepository}$
        If $\boldsymbol{pr}_k$ *[NLA-LA] ==* Q_i
            If $\boldsymbol{pr}_k$ *[LA] ==* $\emptyset$
                Return $\boldsymbol{pr}_k$ *[label]*
            Else if pr_k *[LA] !=* $\emptyset$
                Return (SmartMachine. Result [req_i*])*
}

5- Experimental Results

Based on access histories, a prototype of an access control system is created. The evaluation conditions and datasets utilized are explained first in this section. The evaluation parameters are then set, and the performance of the proposed access control system in the federated learning framework is carefully examined. Finally, the proposed method's performance is compared with previous studies.

5-1- Evaluation Conditions

A system with an Intel Core i7 processor and 16 GB RAM is utilized to analyze the suggested model. Anaconda and Python are used to transfer time series-based learning models from servers to clients and vice versa. We conducted our studies on six datasets and twelve situations. How well the accesses match the original access is one of the major metrics for evaluating access accuracy. In other words, the proposed system's access results are compared to the key policies' access results, and the suggested access control system's quality is evaluated. For example, if the proposed system predicts that an access request will be granted and access authorization is granted in the primary policy, the quality of the suggested system would improve.

5-2- Datasets

We employ six datasets, including real and conditional datasets, to evaluate the performance of the proposed approach. Real datasets such as Kaggle and Amazon UCI datasets [47][48] are employed, as well as conditional datasets. The RESOURCE, MGR ID, ROLE ROLLUP1, ROLE ROLLUP2, ROLE DEPTNAME, ROLE TITLE, ROLE FAMILY DESC, ROLE FAMILY, ROLE CODE, and a label field are among the ten features in the Kaggle dataset.

In addition, the UCI dataset has five features and approximately 716K samples, including ACTION, TARGET NAME, LOGIN, REQUEST DATE, AUTHORIZATION DATE, and labels.
Four conditional datasets are also employed, where each one is generated based on the criteria and sequence of accesses. These four datasets are created using Python and based on the description of numerous requirements. They comprise features such as Name, Role, Time, Location, Sensitivity, and labels. Table 2 shows the general characteristics of the datasets used. D is the name of the dataset used in this table, S# is the number of dataset samples, F# is the number of dataset features, P^+ is the number of samples labeled 1 and P^- is the number of samples labeled zero.

Table 2. Characteristics of the employed datasets

#	D	S#	F#	P^+	p^-
d_K	*Amazon Kaggle*	*32769*	*10*	*30872*	*1897*
$d_{K,1}$ - $d_{K,10}$	*Datasets $d_{K,1}$ - $d_{K,10}$ are created from dataset d_K*	*32769*	*10*	*30872*	*1897*
d_U	*Amazon UCI*	*716063*	*3*	*705152*	*10911*
$d_{U,1} - d_{U,10}$	*Datasets $d_{U,1}$ - $d_{U,10}$ are created from dataset d_U*	*716063*	*3*	*705152*	*10911*
d_{C1}	*Conditional Dataset1*	*10000*	*6*	*9105*	*895*
$d_{C1,1} - d_{C1,10}$	*Datasets $d_{C1,1} - d_{C1,10}$ are created from dataset d_{C1}*	*10000*	*6*	*9105*	*895*
d_{C2}	*Conditional Dataset2*	*10000*	*6*	*7022*	*2978*
$d_{C2,1} - d_{C2,10}$	*Datasets $d_{C3,1} - d_{C3,10}$ are created from dataset d_{C3}*	*10000*	*6*	*7022*	*2978*
d_{C3}	*Conditional Dataset2*	*10000*	*6*	*7022*	*2978*
$d_{C3,1} - d_{C3,10}$	*Datasets $d_{C3,1} - d_{C3,10}$ are created from dataset d_{C3}*	*50000*	*6*	*5888*	*4112*
d_{C4}	*Conditional Dataset4*	*10000*	*6*	*29833*	*20167*
$d_{C4,1} - d_{C4,10}$	*Datasets $d_{C4,1} - d_{C4,10}$ are created from dataset d_{C4}*	*10000*	*6*	*29833*	*20167*

5-3- Results

In the first stage, the performance of the proposed method is investigated; conventional classification models such as KNN, SVM, DT, and NN, as well as time series classification models such as Simple RNN, LSTM, and GRU, have been implemented on datasets d_K, d_U, d_{C1}, d_{C2}, d_{C3}, and d_{C4}, and the results are shown in figures 5 and 6. As can be observed, models based on time series, such as LSTM, outperform conventional classification methods in terms of correct identification of the access control system.

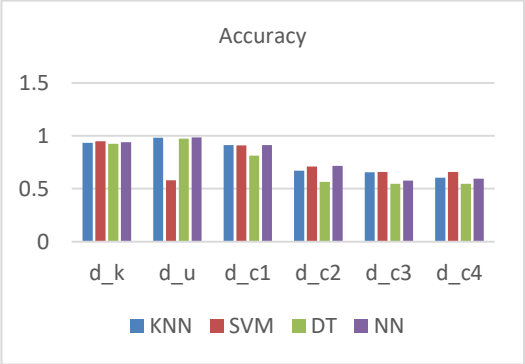

Figure 5. Accuracy of the proposed approach in algorithms without time series in different datasets

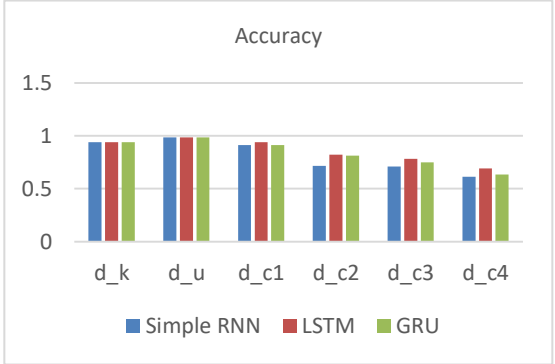

Figure 6. Accuracy of the proposed approach in time series algorithms in different datasets

The performance depicted in the figures above is estimated assuming that all data is accessible. However, the suggested method places the data in local healthcare systems and prevents it from being forwarded to servers. As a result, the dataset d_K is partitioned into ten smaller datasets numbered d_{K1}

through d_{K10}. Similarly, $d_{U,1}$ to $d_{U,10}$, $d_{C1,1}$ to $d_{C1,10}$, $d_{C2,1}$ to $d_{C2,10}$, $d_{C3,1}$ to $d_{C3,10}$, and $d_{C4,1}$ to $d_{C4,10}$ are formed from d_U datasets, as are d_{C1}, d_{C2}, d_{C3}, and d_{C4}. The separated datasets are located in local healthcare systems (Client), and none of these data are stored on the server owing to privacy concerns. On the server, a time series-based learning model is defined. This model is then given to ten clients depending on the written codes. The received model is applied to the data in clients, and the model is trained. The models' output (weight matrix) is then provided to the servers. The results are aggregated on the servers and resent to the clients. Table 3 displays the acquired results.

Table 3. Comparing the accuracy of the trained model in Local HIS systems and aggregated via federated learning architecture

	$partial_1$ DS	$Partial_2$ DS	$Partial_3$ DS	$Partial_4$ DS	$Partial_5$ DS	$Partial_6$ DS	$Partial_7$ DS	$Partial_8$ DS	$Partial_9$ DS	$partial_{10}$ DS	Total DS
d_K	ds=$d_{K,1}$ Acc=0.931	ds=$d_{K,2}$ Acc=0.925	ds=$d_{K,3}$ Acc=0.913	ds=$d_{K,4}$ Acc=0.933	ds=$d_{K,5}$ Acc=0.928	ds=$d_{K,6}$ Acc=0.920	ds=$d_{K,7}$ Acc=0.919	ds=$d_{K,8}$ Acc=0.926	ds=$d_{K,9}$ Acc=0.921	ds=$d_{K,10}$ Acc=0.918	Acc=0.939
d_U	ds=$d_{U,1}$ Acc=0.961	ds=$d_{U,2}$ Acc=0.971	ds=$d_{U,3}$ Acc=0.972	ds=$d_{U,4}$ Acc=0.975	ds=$d_{U,5}$ Acc=0.973	ds=$d_{U,6}$ Acc=0.981	ds=$d_{U,7}$ Acc=0.972	ds=$d_{U,8}$ Acc=0.969	ds=$d_{U,9}$ Acc=0.980	ds=$d_{U,10}$ Acc=0.970	Acc=0.985
d_{C1}	ds=$d_{C1,1}$ Acc=0.913	ds=$d_{C1,2}$ Acc=0.921	ds=$d_{C1,3}$ Acc=0.911	ds=$d_{C1,4}$ Acc=0.933	ds=$d_{C1,5}$ Acc=0.925	ds=$d_{C1,6}$ Acc=0.931	ds=$d_{C1,7}$ Acc=0.937	ds=$d_{C1,8}$ Acc=0.929	ds=$d_{C1,9}$ Acc=0.934	ds=$d_{C1,10}$ Acc=0.928	Acc=0.941
d_{C2}	ds=$d_{C2,1}$ Acc=0.811	ds=$d_{C2,2}$ Acc=0.814	ds=$d_{C2,3}$ Acc=0.797	ds=$d_{C2,4}$ Acc=0.921	ds=$d_{C2,5}$ Acc=0.812	ds=$d_{C2,6}$ Acc=0.792	ds=$d_{C2,7}$ Acc=0.819	ds=$d_{C2,8}$ Acc=0.824	ds=$d_{C2,9}$ Acc=0.810	ds=$d_{C2,10}$ Acc=0.799	Acc=0.823
d_{C3}	ds=$d_{C3,1}$ Acc=0.751	ds=$d_{C3,2}$ Acc=0.743	ds=$d_{C3,3}$ Acc=0.766	ds=$d_{C3,4}$ Acc=0.759	ds=$d_{C3,5}$ Acc=0.739	ds=$d_{C3,6}$ Acc=0.752	ds=$d_{C3,7}$ Acc=0.750	ds=$d_{C3,8}$ Acc=0.744	ds=$d_{C3,9}$ Acc=0.753	ds=$d_{C3,10}$ Acc=0.753	Acc=0.783
d_{C4}	ds=$d_{C4,1}$ Acc=0.622	ds=$d_{C4,2}$ Acc=0.634	ds=$d_{C4,3}$ Acc=0.651	ds=$d_{C4,4}$ Acc=0.957	ds=$d_{C4,5}$ Acc=0.641	ds=$d_{C4,6}$ Acc=0.648	ds=$d_{C4,7}$ Acc=0.636	ds=$d_{C4,8}$ Acc=0.659	ds=$d_{C4,9}$ Acc=0.645	ds=$d_{C4,10}$ Acc=0.660	Acc=0.691

5-4- Comparison with Recent Studies

To demonstrate the outperformance of the proposed approach, its performance is compared with that of Karimi [44] and Cotrini [46]. These methods have calculated the accuracy of their methods using machine learning algorithms. The results show the higher performance of the proposed method compared with others in problems that include time series conditions. Employing the machine learning models based on time series in d_K, d_U, d_{C1}, d_{C2}, d_{C3}, and d_{C4} using federated learning has a higher accuracy compared to the method presented by Karimi and Cotrini, as shown in Figure 7. The accuracy of the suggested model, which is based on time series, is compared with the research of Karimi and Cotrini in each local data corresponding to d_{K1} to d_{K10}, as well as d_{U1} to d_{U10}, $d_{C1,1}$ to $d_{C1,10}$, $d_{C2,1}$ to $d_{C2,10}$, and $d_{C3,1}$ to $d_{C3,10}$, $d_{C4,1}$ to $d_{C4,10}$ and shown in Table 4.

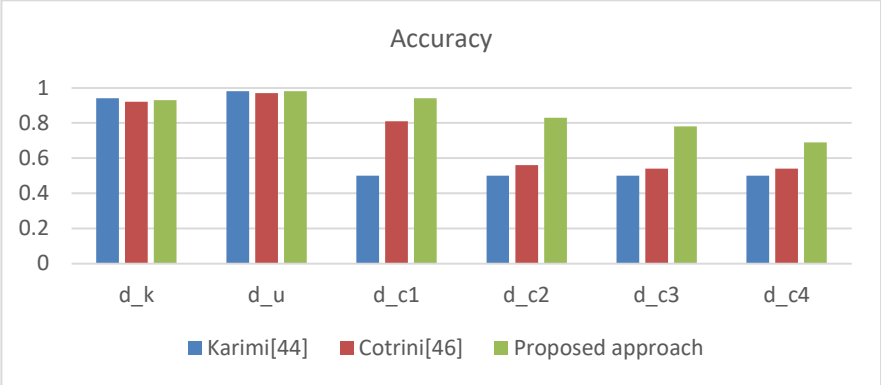

Figure 7. The accuracy comparison of the proposed method with other methods

Table 4. Comparing the accuracy of the trained model in Local HIS systems in each local data

	$partial_1$ DS	$Partial_2$ DS	$Partial_3$ DS	$Partial_4$ DS	$Partial_5$ DS	$Partial_6$ DS	$Partial_7$ DS	$Partial_8$ DS	$Partial_9$ DS	$partial_{10}$ DS	Aggregated DS
d_K *Karimi[44] Cotrini[46] Proposed approach*	$ds=d_{K,1}$ Acc=0.934 Acc=0.845 Acc=0.931	$ds=d_{K,2}$ Acc=0.931 Acc=0.844 Acc=0.925	$ds=d_{K,3}$ Acc=0.93 Acc=0.841 Acc=0.913	$ds=d_{K,4}$ Acc=0.943 Acc=0.829 Acc=0.933	$ds=d_{K,5}$ Acc=0.948 Acc=0.839 Acc=0.928	$ds=d_{K,6}$ Acc=0.937 Acc=0.839 Acc=0.920	$ds=d_{K,7}$ Acc=0.939 Acc=0.840 Acc=0.919	$ds=d_{K,8}$ Acc=0.946 Acc=0.829 Acc=0.936	$ds=d_{K,9}$ Acc=0.941 Acc=0.824 Acc=0.921	$ds=d_{K,10}$ Acc=0.938 Acc=0.834 Acc=0.918	Acc=0.943 Acc=0.849 Acc=0.939
d_U *Karimi[44] Cotrini[46] Proposed approach*	$ds=d_{U,1}$ Acc=0.961 Acc=0.673 Acc=0.961	$ds=d_{U,2}$ Acc=0.971 Acc=0.687 Acc=0.971	$ds=d_{U,3}$ Acc=0.965 Acc=0.671 Acc=0.972	$ds=d_{U,4}$ Acc=0.970 Acc=0.693 Acc=0.975	$ds=d_{U,5}$ Acc=0.975 Acc=0.682 Acc=0.973	$ds=d_{U,6}$ Acc=0.984 Acc=0.679 Acc=0.981	$ds=d_{U,7}$ Acc=0.972 Acc=0.691 Acc=0.972	$ds=d_{U,8}$ Acc=0.979 Acc=0.688 Acc=0.965	$ds=d_{U,9}$ Acc=0.976 Acc=0.682 Acc=0.980	$ds=d_{U,10}$ Acc=0.965 Acc=0.689 Acc=0.970	Acc=0.984 Acc=0.704 Acc=0.982
d_{C1} *Karimi[44] Cotrini[46] Proposed approach*	$ds=d_{C1,1}$ Acc=0.551 Acc=0.480 Acc=0.913	$ds=d_{C1,2}$ Acc=0.524 Acc=0.488 Acc=0.921	$ds=d_{C1,3}$ Acc=0.537 Acc=0.472 Acc=0.911	$ds=d_{C1,4}$ Acc=0.542 Acc=0.485 Acc=0.933	$ds=d_{C1,5}$ Acc=0518 Acc=0.49 Acc=0.925	$ds=d_{C1,6}$ Acc=0.543 Acc=0.491 Acc=0.931	$ds=d_{C1,7}$ Acc=0.517 Acc=0.47 Acc=0.937	$ds=d_{C1,8}$ Acc=0.498 Acc=0.492 Acc=0.929	$ds=d_{C1,9}$ Acc=0.519 Acc=0.486 Acc=0.934	$ds=d_{C1,10}$ Acc=0.523 Acc=0.487 Acc=0.928	Acc=0.556 Acc=0.501 Acc=0.941
d_{C2} *Karimi[44] Cotrini[46] Proposed approach*	$ds=d_{C2,1}$ Acc=0.475 Acc=0.459 Acc=0.811	$ds=d_{C2,2}$ Acc=0.490 Acc=0.461 Acc=0.814	$ds=d_{C2,3}$ Acc=0.494 Acc=0.469 Acc=0.797	$ds=d_{C2,4}$ Acc=0.489 Acc=0.471Acc=0.921	$ds=d_{C2,5}$ Acc=0.472 Acc=0.472 Acc=0.812	$ds=d_{C2,6}$ Acc=0.499 Acc=0.457 Acc=0.792	$ds=d_{C2,7}$ Acc=0.491 Acc=0.479 Acc=0.819	$ds=d_{C2,8}$ Acc=0.482 Acc=0.466 Acc=0.824	$ds=d_{C2,9}$ Acc=0.484 Acc=0.463 Acc=0.810	$ds=d_{C2,10}$ Acc=0.497 Acc=0.478 Acc=0.799	Acc=0.521 Acc=0.482 Acc=0.823
d_{C3} *Karimi[44] Cotrini[46] Proposed approach*	$ds=d_{C3,1}$ Acc=0.489 Acc=0.392 Acc=0.751	$ds=d_{C3,2}$ Acc=0.494 Acc=0.387 Acc=0.743	$ds=d_{C3,3}$ Acc=0.487 Acc=0.399 Acc=0.766	$ds=d_{C3,4}$ Acc=0.492 Acc=0.404 Acc=0.759	$ds=d_{C3,5}$ Acc=0492 Acc=0.385 Acc=0.739	$ds=d_{C3,6}$ Acc=0.501 Acc=0.390 Acc=0.752	$ds=d_{C3,7}$ Acc=0.497 Acc=0.399 Acc=0.750	$ds=d_{C3,8}$ Acc=0.498 Acc=0.385 Acc=0.744	$ds=d_{C3,9}$ Acc=0.489 Acc=0.409 Acc=0.753	$ds=d_{C3,10}$ Acc=0.493 Acc=0.396 Acc=0.753	Acc=0.503 Acc=0.413 Acc=0.783

d_{C2}	ds=$d_{C2,1}$	ds=$d_{C2,2}$	ds=$d_{C2,3}$	ds=$d_{C2,4}$	ds=$d_{C2,5}$	ds=$d_{C2,6}$	ds=$d_{C2,7}$	ds=$d_{C2,8}$	ds=$d_{C2,9}$	ds=$d_{C2,10}$	
Karimi[44]	Acc=0.466	Acc=0.580	Acc=0.457	Acc=0.455	Acc=0.460	Acc=0.471	Acc=0.484	Acc=0.472	Acc=0.472	Acc=0.457	Acc=0.489
Cotrini[46]	Acc=0.377	Acc=0.386	Acc=0.392	Acc=0.389	Acc=0.374	Acc=0.367	Acc=0.390	Acc=0.387	Acc=0.375	Acc=0.391	Acc=0.401
Proposed approach	Acc=0.622	Acc=0.634	Acc=0.651	Acc=0.957	Acc=0.641	Acc=0.648	Acc=0.636	Acc=0.659	Acc=0.645	Acc=0.660	Acc=0.691

6- Discussion

The proposed approach's performance has been carefully investigated in six datasets, and it can be observed that the accuracy of the proposed access control works well in d_K and d_U datasets using traditional machine learning methods such as SVM, but in conditional datasets, regardless of the circumstances, because access is complicated depending on logs, the techniques outlined above have poorer accuracy in granting right permission. Meanwhile, with more complex conditions in d_{C1} to d_{C4} datasets, the suggested method employing deep recurrent networks, particularly the LSTM network, outperforms other algorithms. Another advantage of the proposed method is that it provides internet access in a short time, which is due to the use of hidden memory in the recurrent neural network, as well as the training of this network during each access and testing the network at the time of access request so that no additional burden is imposed on the access control system when a new request is received. Furthermore, using federated learning architecture protects user privacy in HIS systems and integrates knowledge from other systems to allow access to users.

7- Conclusion

In this paper, a new approach for granting access authorization in healthcare systems online based on intelligent module decisions was provided. The utilized module responds to access requests using time series learning models such as LSTM and GRU. Local data from healthcare systems cannot also be aggregated in a single dataset. As a result, federated learning architecture and machine learning algorithms were remotely applied to various healthcare systems. The servers were in charge of gathering various learnings and relaying them to the local systems to regulate access based on the experiences of all systems. The experimental results reveal that the performance of the access control system in local systems after implementing the federated learning architecture and aggregating the knowledge of local systems is lower than the performance of this system before implementing the proposed approach.

References

[1] Haux, R. Health information systems–past, present, future. International journal of medical informatics, 75(3-4), 268-281, 2006.

[2] Ravidas, S., Lekidis, A., Paci, F., & Zannone, N. Access control in Internet-of-Things: A survey. Journal of Network and Computer Applications, 144, 79-101, 2019.

[3] Ding, S., Cao, J., Li, C., Fan, K., & Li, H. A novel attribute-based access control scheme using blockchain for IoT. IEEE Access, 7, 38431-38441, 2019.

[4] Hu, V. C., Kuhn, D. R., Ferraiolo, D. F., & Voas, J. Attribute-based access control. Computer, 48(2), 85-88, 2015.

[5] Wouters, O. J., Shadlen, K. C., Salcher-Konrad, M., Pollard, A. J., Larson, H. J., Teerawattananon, Y., & Jit, M. Challenges in ensuring global access to COVID-19 vaccines: production, affordability, allocation, and deployment. The Lancet, 397(10278), 1023-1034, 2021.

[6] Hu, V. C., Ferraiolo, D., Kuhn, R., Friedman, A. R., Lang, A. J., Cogdell, M. M., ... & Scarfone, K. Guide to attribute-based access control (abac) definition and considerations (draft). NIST special publication, 800(162), 1-54, 2013.

[7] Zaremba, W., Sutskever, I., & Vinyals, O. Recurrent neural network regularization. arXiv preprint arXiv:1409.2329, 2014.

[8] Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306, 2020.

[9] Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. Recent advances in recurrent neural networks. arXiv preprint arXiv:1801.01078, 2017.

[10] Lipton, Z. C., Berkowitz, J., & Elkan, C. A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019, 2015.

[11] Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., & Gao, Y. A survey on federated learning. Knowledge-Based Systems, 216, 106775, 2021.

[12] Aledhari, M., Razzak, R., Parizi, R. M., & Saeed, F. Federated learning: A survey on enabling technologies, protocols, and applications. IEEE Access, 8, 140699-140725, 2020.

[13] Dhanvijay, M. M., & Patil, S. C. Internet of Things: A survey of enabling technologies in healthcare and its applications. Computer Networks, 153, 113-131, 2019.

[14] Alam, M. M., Malik, H., Khan, M. I., Pardy, T., Kuusik, A., & Le Moullec, Y. A survey on the roles of communication technologies in IoT-based personalized healthcare applications. IEEE Access, 6, 36611-36631, 2018.

[15] Wang, H., Zhang, X., Xia, Y., & Wu, X. An intelligent blockchain-based access control framework with federated learning for genome-wide association studies. Computer Standards & Interfaces, 84, 103694, 2023.
[16] Shojafar, M., Mukherjee, M., Piuri, V., & Abawajy, J. Guest editorial: Security and privacy of federated learning solutions for industrial IoT applications. IEEE Transactions on Industrial Informatics, 18(5), 3519-3521, 2021.
[17] Mazzocca, C., Romandini, N., Colajanni, M., & Montanari, R. FRAMH: A Federated Learning Risk-Based Authorization Middleware for Healthcare. IEEE Transactions on Computational Social Systems, 2022.
[18] Bhansali, P. K., Hiran, D., Kothari, H., & Gulati, K. Cloud-based secure data storage and access control for the internet of medical things using federated learning. International Journal of Pervasive Computing and Communications, (ahead-of-print), 2022.
[19] Dhiman, G., Juneja, S., Mohafez, H., El-Bayoumy, I., Sharma, L. K., Hadizadeh, M., ... & Khandaker, M. U. Federated learning approach to protect healthcare data over big data scenario. Sustainability, 14(5), 2500, 2022.
[20] Jabal, A. A., Bertino, E., Lobo, J., Verma, D., Calo, S., & Russo, A. FLAP--A Federated Learning Framework for Attribute-based Access Control Policies. arXiv preprint arXiv:2010.09767, 2020.
[21] Ghimire, B., & Rawat, D. B. Recent advances on federated learning for cybersecurity and cybersecurity for federated learning for internet of things. IEEE Internet of Things Journal, 2022.
[22] Savazzi, S., Nicoli, M., Bennis, M., Kianoush, S., & Barbieri, L. Opportunities of federated learning in connected, cooperative, and automated industrial systems. IEEE Communications Magazine, 59(2), 16-21, 2021.
[23] Geng, J., Kanwal, N., Jaatun, M. G., & Rong, C. DID-eFed: Facilitating Federated Learning as a Service with Decentralized Identities. In Evaluation and Assessment in Software Engineering (pp. 329-335), 2021.
[24] Alam, T., & Gupta, R. Federated Learning and Its Role in the Privacy Preservation of IoT Devices. Future Internet, 14(9), 246, 2022.
[25] Kim, T. Y., & Cho, S. B. Optimizing CNN-LSTM neural networks with PSO for anomalous query access control. Neurocomputing, 456, 666-677, 2021.
[26] Ye, X., Yu, Y., & Fu, L. Multi-Channel Opportunistic Access for Heterogeneous Networks Based on Deep Reinforcement Learning. IEEE Transactions on Wireless Communications, 21(2), 794-807, 2021.
[27] Otoum, Y., Liu, D., & Nayak, A. DL-IDS: a deep learning–based intrusion detection framework for securing IoT. Transactions on Emerging Telecommunications Technologies, 33(3), e3803, 2022.
[28] Kumar, A., Abhishek, K., Bhushan, B., & Chakraborty, C. Secure access control for manufacturing sector with application of ethereum blockchain. Peer-to-Peer Networking and Applications, 14(5), 3058-3074, 2021.
[29] Zhong, H., Zhou, Y., Zhang, Q., Xu, Y., & Cui, J. An efficient and outsourcing-supported attribute-based access control scheme for edge-enabled smart healthcare. Future Generation Computer Systems, 115, 486-496, 2021.
[30] Kumar, R., & Tripathi, R. Scalable and secure access control policy for healthcare system using blockchain and enhanced Bell–LaPadula model. Journal of Ambient Intelligence and Humanized Computing, 12(2), 2321-2338, 2021.
[31] Egala, B. S., Pradhan, A. K., Badarla, V., & Mohanty, S. P.Fortified-chain: a blockchain-based framework for security and privacy-assured internet of medical things with effective access control. IEEE Internet of Things Journal, 8(14), 11717-11731, 2021.
[32] Singh, A., & Chatterjee, K. LoBAC: A Secure Location-Based Access Control Model for E-Healthcare System. In Advances in Machine Learning and Computational Intelligence (pp. 621-628). Springer, Singapore, 2021.
[33] Younis, M., Lalouani, W., Lasla, N., Emokpae, L., & Abdallah, M. Blockchain-enabled and data-driven smart healthcare solution for secure and privacy-preserving data access. IEEE Systems Journal, 2021.
[34] Ghayvat, H., Pandya, S., Bhattacharya, P., Zuhair, M., Rashid, M., Hakak, S., & Dev, K. CP-BDHCA: Blockchain-based Confidentiality-Privacy preserving Big Data scheme for healthcare clouds and applications. IEEE Journal of Biomedical and Health Informatics, 26(5), 1937-1948, 2021.
[35] Yaqoob, I., Salah, K., Jayaraman, R., & Al-Hammadi, Y. Blockchain for healthcare data management: opportunities, challenges, and future recommendations. Neural Computing and Applications, 34(14), 11475-11490, 2022.
[36] Azad, M. A., Arshad, J., Mahmoud, S., Salah, K., & Imran, M.. A privacy-preserving framework for smart context-aware healthcare applications. Transactions on Emerging Telecommunications Technologies, 33(8), e3634, 2022.
[37] Balaji, N. V. An attack Resistant Privacy-Preserving Access Control Scheme for Outsourced E-pharma Data in Cloud. International Journal of Next-Generation Computing, 13(3), 2022.
[38] Tao, Q., & Cui, X. B-FLACS: blockchain-based flexible lightweight access control scheme for data sharing in cloud. Cluster Computing, 1-11, 2022.
[39] Pal, S., Dorri, A., & Jurdak, R. Blockchain for IoT access control: Recent trends and future research directions. Journal of Network and Computer Applications, 103371, 2022.
[40] Ghillani, D. Deep Learning and Artificial Intelligence Framework to Improve the Cyber Security. Authorea Preprints, 2022.
[41] Chinnasamy, P., & Deepalakshmi, P. HCAC-EHR: hybrid cryptographic access control for secure EHR retrieval in healthcare cloud. Journal of Ambient Intelligence and Humanized Computing, 13(2), 1001-1019, 2022.
[42] Astillo, P. V., Duguma, D. G., Park, H., Kim, J., Kim, B., & You, I. Federated intelligence of anomaly detection agent in IoTMD-enabled Diabetes Management Control System. Future Generation Computer Systems, 128, 395-405, 2022.
[43] Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., ... & He, B. A survey on federated learning systems: vision, hype and reality for data privacy and protection. IEEE Transactions on Knowledge and Data Engineering, 2021.
[44] Karimi, L., Aldairi, M., Joshi, J., & Abdelhakim, M. An automatic attribute based access control policy extraction from access logs. IEEE Transactions on Dependable and Secure Computing, 2021.
[45] Hu, V. Machine Learning for Access Control Policy Verification (No. NIST Internal or Interagency Report (NISTIR) 8360 (Draft)). National Institute of Standards and Technology, 2021.

[46] Cotrini, C., Weghorn, T., & Basin, D. Mining ABAC rules from sparse logs. In 2018 IEEE European Symposium on Security and Privacy (EuroS&P) (pp. 31-46). IEEE, 2018.
[47] Amazon.com, "Amazon employee access challenge." Kaggle.
[48] Montanez, Ken, "Amazon access samples." UCI Machine Learning Repository: Amazon Access Samples Data Set.
[49] Rikhtechi, L., Rafe, V., & Rezakhani, A. Secured access control in security information and event management systems. Journal of Information Systems and Telecommunication, 9(33), 67-78, 2021.
[50] Rathna, R., Gladence, L. M., Cynthia, J. S., & Anu, V. M Energy efficient cross layer MAC protocol for wireless sensor networks in remote area monitoring applications. Journal of Information Systems and Telecommunication (JIST), 3(35), 207, 2021.