CASE: A Context-Aware Security Scheme for Preserving Data Privacy in IoT-Enabled Society 5.0

Timam Ghosh, Arijit Roy, *Member, IEEE*, Sudip Misra, *Senior Member, IEEE*, and Narendra Singh Raghuwanshi

***Abstract*—This article introduces the concept of context-aware attribute learning with cipher policy-attribute-based encryption (CP-ABE) to preserve the privacy of users' information in IoT-enabled Society 5.0. The concept of Society 5.0 pioneers an abstract system unifying different smart environments (SEs) to provide seamless services to the citizens. While serving different applications, these SEs store users' information in the cloud engendering users' privacy. CP-ABE is one of the conventional security systems that preserves privacy with group data accessibility. Contemporary CP-ABE solutions enforce users to manually provide their contextual information, namely, attributes, to encrypt/decrypt data. From these solutions it can be conjectured that incorrect attribute selection by a user raises the issue of unauthenticated access to information. To address these issues, we propose a scheme, named the context-aware attribute learning scheme (CASE), which autonomously learns users' contextual information, exploiting edge intelligence, generates attributes, and reduces the post-encryption data size using the learned attributes. We examine the performance of CASE with the help of a case study on CP-ABE over smart healthcare systems (SHSs). Extensive experimental results show that CASE outperforms the existing CP-ABE-based security schemes by reducing 32%–33% average network delay, 33%–35% average energy consumption, and 31%–36% average packet loss. Additionally, we analyze the performance of attribute learning schemes using the support vector machine (SVM), decision tree (DT), and naive Bayes (NB) learning models. We observe that DT reports better performance over SVM and NB in prediction accuracy, prediction time, and clock cycles required for execution.**

***Index Terms*—Attribute learning, context-aware information, cipher policy-attribute-based encryption (CP-ABE), edge computing, Internet of Things (IoT), machine learning, Society 5.0.**

I. Introduction

Society 5.0 [1] ideates to unify different smart environments (SEs) such as smart healthcare, smart home automation, smart transport, and smart industry by integrating artificial intelligence (AI), Edge Intelligence, and Internet of Things (IoT) [1]. Typically, these SEs deal with users' personal information collected from edge devices using IoT sensors, which are archived in a third-party server or cloud for future analysis and utilization. Storing information in the cloud/server raises the demand for maintaining the privacy of these users' information [2], [3]. The existing security solutions [4]–[7] preserve the privacy of these information by encrypting them using the private and public keys. These solutions primarily focus on reliable key generation, while increasing data accessing time. Therefore, SEs in Society 5.0 need a security solution with suitable data accessibility features to preserve the privacy of users' information. However, attribute-based encryption (ABE) [6] reduces the overhead in accessing data from the cloud, while preserving data privacy. ABE uses paired keys to encrypt the information before storing it in the cloud. It uses group accessibility with users' contextual information or attributes. Moreover, ABE oversees the key generation distribution using attribute authority (AA) [7].

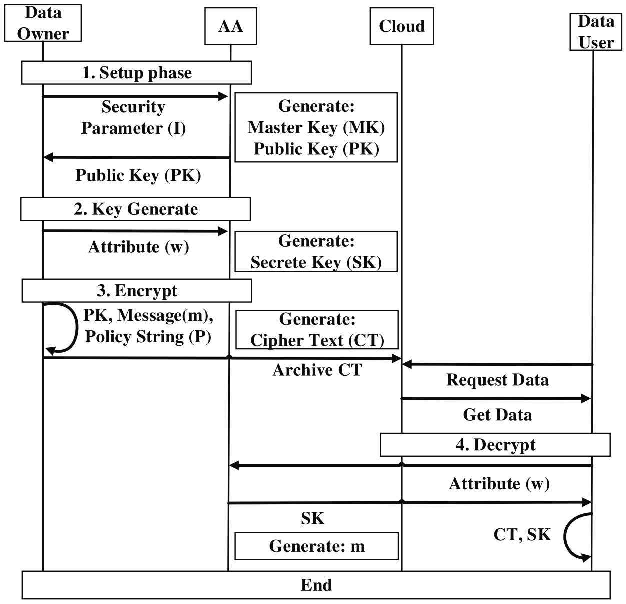

Fig. 1. Data flow diagram of existing CP-ABE.

This article considers a cipher policy-ABE (CP-ABE)-based security system [8] for preserving data privacy in IoT-enabled Society 5.0. As a case study, we consider a smart healthcare system (SHS) with CP-ABE, for which a generic data flow diagram to preserve data privacy [8] is depicted in Fig. 1. This figure shows that CP-ABE combines information with

Manuscript received January 1, 2021; revised June 12, 2021; accepted July 12, 2021. Date of publication July 29, 2021; date of current version February 4, 2022. *(Corresponding author: Sudip Misra.)*

Timam Ghosh and Arijit Roy are with the Advanced Technology Development Centre, Indian Institute of Technology Kharagpur, Kharagpur 721302, India (e-mail: timam.ghosh@iitkgp.ac.in; arijitroy@iitkgp.ac.in).

Sudip Misra is with the Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, India (e-mail: smisra@cse.iitkgp.ac.in).

Narendra Singh Raghuwanshi is with the Department of Agricultural and Food Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, India (e-mail: nsr@agfe.iitkgp.ernet.in).

Digital Object Identifier 10.1109/JIOT.2021.3101115

the access policy and keys with contextual information or attributes during the encryption process. While implementing CP-ABE, the existing systems face the following issues.

1) The existing systems enforce the users to provide their attributes manually, which are used to restrict the data access. The incorrect selection of the attributes by the user raises the issue of unauthenticated information access. In the context of SHS, a patient provides "laying" as an attribute to encrypt corresponding activity-based parameters and generate the access policy: <doctors AND laying>. If the patients encrypt the same information with attributes other than laying, any user other than the doctor can access the information, which is undesirable.
2) The existing security system [4] reports the issue of increasing the post-encryption data size [6], [7], [9]. Similarly, CP-ABE [8], being a variant of such a system, inherits the same issue of increasing the data size during the post-encryption phase. Due to the rapidly growing IoT devices, the issue of increasing post-encryption data size in CP-ABE creates a bottleneck while archiving the data at the cloud. Consequently, it elevates the issue of increased packet loss, latency, and energy consumption.

The aforementioned issues significantly reduce the efficiency of the SEs in Society 5.0 and affect the normal operation in real-time applications. Therefore, it is pertinent to design a context-aware CP-ABE-based security scheme for Society 5.0, which is automatically capable of learning the users' attributes and reduce the post-encryption data size.

In this work, we propose a context-aware attribute learning scheme, named CASE, which is capable of providing context-aware security by exploiting edge intelligence in IoT-enabled Society 5.0. The specific *contributions* of this work are as follows.

1) In the existing approaches in CP-ABE, the users manually provide the attributes, which produces the issue of unauthenticated data access. To address this issue, we propose CASE—an attribute learning scheme, that automatically learns users' activities as context or attributes in edge devices for encrypting the user information.
2) In the proposed scheme, CASE, we design a mechanism that adopts the functionality of CP-ABE. We consider SHS as a case study for implementing the proposed scheme. In this system, the user/edge devices collect their information using smart sensors, encrypt them, and transmit the encrypted information to the cloud using the access points. The system uses fog devices to perform the functionality of AA, which involves generating and distributing the public and private keys.
3) For reducing the size of the post-encrypted data in the existing approaches, we introduced an attribute learning scheme in CASE that extracts users' contextual information from a portion of the data using a pretrained learning model. CASE encapsulates this contextual information in the attributes, which are used during the data encryption and decryption phases. The scheme encrypts the remaining portion of the data using public key and policy string and transmits them to the remote server without losing its information.
4) For extracting user's contextual information in CASE, we explore different learning models, which are pretrained using the support vector machine (SVM), decision tree (DT), and naive Bayes (NB). We consider these lightweight machine learning approaches to enable users' context prediction in the edge device. Furthermore, we characterize CASE with theoretical analysis and study the performance of CASE with rigorous simulation.

II. Related Works

This section discusses the existing research works on CP-ABE-based security systems. To provide real-time and seamless IoT services to the citizens, Society 5.0 needs to enforce rules [2] for preserving users' privacy, while storing users' personal information in the cloud for future analysis. ABE [6], [7] becomes a popular security tool that offers privacy with efficient data accessibility in a system. To provide information privacy, Susilo *et al.* [6] and Odelu *et al.* [7] proposed a threshold-based ABE without dummy attribute to generate a fixed-sized ciphertext and a pairing-based CP-ABE to generate a fixed-size ciphertext and secrete key, respectively. On the other hand, Rana and Mishra [10] and Sojanya *et al.* [11] proposed an efficient and security preserving scheme to address the vulnerability of existing SHS due to insufficient security proofs and Man-in-the-middle attack. Khan *et al.* [12] introduces a patient-centric SHS involving a multimessage CPABE scheme and efficient policy update to alleviate the issue of static policy update in state-of-the-art systems. On the other hand, Saravanan and Umamakeswari [13] proposed a CP-ABE-based cryptosystem using the HAP authentication scheme for SHS to adopt huge security parameter requirements in existing CP-ABE-based SHS. These CP-ABE solutions for SHSs suggest mechanisms to reduce the size of post encrypted data during the encryption process and statically select attributes for encrypting the EHR, keeping human-in-the-loop. However, these security systems need an autonomous attribute selection mechanism for eliminating unauthenticated data access due to static attribute selection in Society 5.0.

Khalifa *et al.* [14] proposed HARKE—a green human activity recognition model using piezoelectric energy harvester and wearable device. Bharti *et al.* [15] used multimodel and multipositional wearable sensors to recognize at-home human activities. On the other hand, Casella *et al.* [16] used the same devices to recognize outdoor human activities and proposed a finite-state automata-based human activity recognition model. These approaches recognize various human activities, such as running, walking, lying on the bed, sofa, or floor, walking upstairs, and walking downstairs. In a similar context, Janarthanan *et al.* [17] proposed an unsupervised human activity recognition using deep learning approaches and WISDOM laboratory data sets that involve data collected from the in-built sensors in various smart devices. Using a similar type of data sets, Dua *et al.* [18] designed their novel feature extraction and human activity classification system using a convolution neural network and gated recurrent unit. Based on the activities, we can categorize the health personnel, who use the patients' information, and provide privacy accordingly. As this

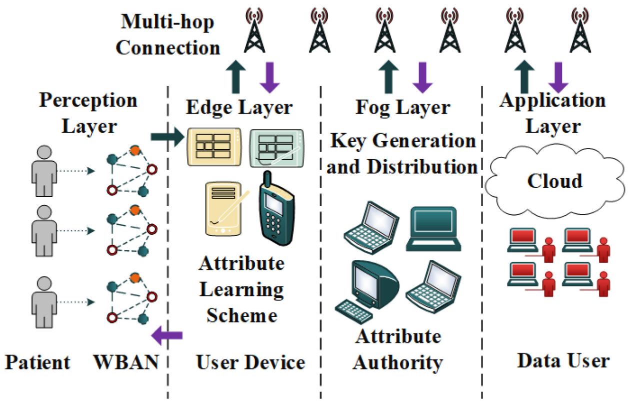

Fig. 2. CP-ABE-enabled SHS in Society 5.0.

information is stored in third-party storage, such as the cloud, the government enforces rules [2] for preserving the privacy of these information. ABE [6], [7] becomes a popular privacy tool that offers efficient data accessibility in a system.

Synthesis: In the existing literature, different authors addressed the issues in the CP-ABE-based SHSs. We observe that the existing approaches (e.g., [6] and [7]) are inefficient in addressing the issues of manual attribute selection by the users. Consequently, these approaches are unable to control unauthenticated data access. Moreover, the existing approaches [6] and [7] increase post-encryption data size, which results in network delay, overall energy consumption, and packet drop.

III. System Model

We propose a cryptosystem in Society 5.0 to render a CP-ABE-based information privacy scheme and model a network opted from the network architecture of Misra and Saha [19]. Fig. 2 depicts a CP-ABE-based system over a smart healthcare environment, which consists of four layers:

1) perception;
2) edge;
3) fog;
4) application.

The perception layer monitors remote patients using wearable sensors and temporarily stores the collected information to the edge device in the edge layer. For preserving patient privacy, the edge devices encrypt the information using CP-ABE before archiving it in a cloud. The patients and other users such as doctors, dietitians, and physical instructors can access the information from the cloud analyzing in detail. CP-ABE uses a trusted fog device [20] in the fog layer to perform the roles of an AA. The proposed system incorporates the concept of a multi-AA security system.

A. *Network Model*

We mode the network as a graph $\mathcal{G}\langle\mathcal{V}, \mathcal{L}\rangle$, $\mathcal{V}$ and $\mathcal{L}$, respectively, denote the set of devices and the set of links. $|\mathcal{V}|$ represents the number of devices, which is categorized into disjoint subsets of:

1) sensor nodes ($\mathcal{V}^s$);
2) edge devices ($\mathcal{V}^e$);
3) fog devices ($\mathcal{V}^f$);
4) access points ($\mathcal{V}^{ap}$);
5) remote servers ($\mathcal{V}^c$).

In $\mathcal{G}$, we consider $|\mathcal{V}^f| = 1$, $\mathcal{V}^f \lll \mathcal{V}^f$, and $|\mathcal{V}^f \cup \mathcal{V}^f \cup \mathcal{V}^f| \ggg \mathcal{V}^{ap}$. ω_i and μ_i are the CPU frequency and the service rate of $v_i \in \mathcal{V}$, respectively.

Definition 1: $f(\cdot)$ denotes the association among the devices.

Definition 2: $\mathcal{V}^p$ denotes peripheral devices, *s.t.* $\mathcal{V}^p = \mathcal{V}^e \cup \mathcal{V}^f \cup \mathcal{V}^c$.

Property 1: Based on Definitions 1 and 2, we illustrate different properties of association among $\mathcal{V}^s$, $\mathcal{V}^{ap}$, $\mathcal{V}^p$and $\mathcal{V}^e$ as follows.

1) $f : \mathcal{V}^s \rightarrow \mathcal{V}^e$, $f : \mathcal{V}^e \rightarrow \mathcal{V}^f$, and $f : \mathcal{V}^p \rightarrow \mathcal{V}^{ap}$ produce one-to-one mapping. Whereas, $f^{-1} : \mathcal{V}^s \leftarrow \mathcal{V}^e$, $f^{-1} : \mathcal{V}^e \leftarrow \mathcal{V}^f$, and $f^{-1} : \mathcal{V}^p \leftarrow \mathcal{V}^{ap}$) produce many-to-one.
2) Each access point in the network can be associated with multiple access points, which produces many-to-many mapping ($f : \mathcal{V}^{ap} \rightarrow \mathcal{V}^{ap}$).

Property 1 is used during the simulating of the system. χ_{ij} is defined as the association variable among the ith and jth devices. $\chi_{ij} = 1$, if $f(v_i) = v_j \forall v_i, v_j \in \mathcal{V}$); otherwise, 0.

Theorem 1: χ_{ij} satisfies $\sum_{j=0}^{|\mathcal{V}^f|} \chi_{ij} \in \{0, 1\}$, $\sum_{i=0}^{|\mathcal{V}^e|} \chi_{ij} \geq 1$.

Proof: Considering the theory of contradiction, we state that Theorem 1 is not true, which reports $\sum_{j=0}^{|\mathcal{V}^f|} \chi_{ij} \geq 1$ and $\sum_{i=0}^{|\mathcal{V}^e|} \chi_{ij} = 1$. Thus, each edge device in the model is associated with multiple fog devices, and $f^{-1}(v_j) = f^{-1}(v_k) = v_i$, where $\forall v_i \in \mathcal{V}^e$ and $\forall v_j, v_k \in \mathcal{V}^e$. The assumption states that the association between the set of edge and fog devices produces one-to-many mapping, which contradicts Property 1. Again, the same assumption states that each fog device can be associated with an edge device and, thus, $f(v_i) \neq f(v_k) = v_j$, where $\forall v_i, v_k \in \mathcal{V}^e$ and $\forall v_j, v_k \in \mathcal{V}^e$. This assumption states that this association produces one-to-one mapping that contradicts Property 1. Thus, from the above discussion, we see that Theorem 1 is true. ■

We denote $\mathcal{L}_{ij} \subset \mathcal{L}$ as the links that represent the path between the ith and the jth devices. Additionally, we define p_{ij}^l as a link variable for the path between the ith and the jth devices, where $p_{ij}^l = 1$ if $\forall l \in \mathcal{L}_{ij}$; otherwise, 0. We denote Δ_l as the distance of the link $l \in \mathcal{L}$.

Theorem 2: Let us assume that p_{ij}^l denote the link, $l \in \mathcal{L}_{ij}$, being used to establish communication between the ith edge and the jth fog devices. Therefore, p_{ij}^l supports $\sum_{\forall l \in \mathcal{L}_{ij}} p_{ij}^l \geq 2$.

Proof: We evaluate Theorem 2 using the theory of contradiction. We assume that Theorem 2 is not true, which reports $\sum_{\forall l \in \mathcal{L}_{ij}} p_{ij}^l \in \{0, 1\}$. Based on this assumption, we can infer two possibilities.

1) $v_i = v_j$, where $\sum_{\forall l \in \mathcal{L}_{ij}} p_{ij}^l = 0$.
2) l_{ij} exists, where $l_{ij} \in \mathcal{L}$ denotes the link between the ith edge and the jth fog devices, and $\sum_{\forall l \in \mathcal{L}_{ij}} p_{ij}^l = 0$.

As discussed previously, $\mathcal{V}^e \cap \mathcal{V}^f = \phi$, which states that $v_i \neq v_j$. Therefore, the possibility cannot be true. Again, as discussed previously, the peripheral devices communicate between themselves via access point-based multihop connections. As a result, l_{ij} cannot exist. From the above discussion, we infer that the assumption is false. ■

B. Task Model

As mentioned previously, the proposed system uses CP-ABE to establish secure communication among the devices that include three main tasks of CP-ABE:

1) key generation and distribution;
2) encryption;
3) decryption.

. In traditional CP-ABE, AA takes the responsibility of key generation and distribution. We introduce a task model, where the ith edge device encrypts and decrypts ∇_i data using $E(\cdot)$ and $D(\cdot)$ functions, respectively. Each device evaluates its data size and CPU cycle requirement using $S(\cdot)$ and $\Omega(\cdot)$ functions, respectively. We consider that each AA generates keys using data with constant size (∇) and requires $\Omega(\nabla)$ CPU cycles.

C. Delay Model

The proposed system involves two types of delays during two different phases—delay for encryption (δ^e) and delay for decryption (δ^d). For modeling the delays, we adopt the delay model used by Misra and Saha [19], which involves:

1) *transmission delay;*
2) *propagation delay;*
3) *queuing delay;*
4) *processing delay.*

1) Transmission Delay: For modeling transmission delay, we consider the log-distance path loss between the ith and the jth devices. This path loss is evaluated as $PL^{ij}_{[dB]} = a + 36.7\log_{10}(\sum_l^{|\mathcal{L}(t)|} p^l_{in} D_l)$, where $a = 140.7 + \mathcal{N}(8)$. Furthermore, we calculate the data rate between these devices as $r_{ij} = B\log_2(1 + (p_i^{tx} - PL^{in}_{[dB]})/\sigma^2)$, where B, p_i^{tx}, and σ^2 denote the link bandwidth, transmission power, and noise power, respectively. Thus, the transmission delay for the ith edge device during encryption is represented as: $\delta^{Tx}_{in} = [S(E(\nabla_i))/r_{in}]$. However, during the decryption process, the ith edge device fetches data from the cloud and requests and receives the secret key from the Fog device. We consider that the transmission delay for fetching data from the remote server and fetching keys from the fog device are negligible.

2) Propagation Delay: We denote δ^l as the propagation delay of the link $l \in \mathcal{L}$. Considering p^l_{ij} as the shortest path between the ith and the jth devices, the total propagation delay between these devices is represented as $\delta^P_{ij} = \sum_l^{|\mathcal{L}(t)|} p^l_{nj}\Delta^l$. During the encryption, the ith edge device transmits the encrypted data to the cloud $\mathcal{V}^c$ through $f^{ap}(v_i)$ access points, and thus, we need to evaluate $\delta^P_{f^{ap}(v_i)c}$. However, during decryption, the ith edge device needs to request and fetch secret key from the AA $f^f(v_i)$ and fetch data from cloud. Thus, we need to evaluate $\delta^P_{f^{ap}(v_i)f^f(v_i)}$, $\delta^P_{f^{ap}(f^f(v_i))v_i}$, and $\delta^P_{f^{ap}(\mathcal{V}^c)v_i}$

$$\delta^E = \frac{\Omega(E(\nabla_i))}{\omega_i} + \frac{S(E(\nabla_i))}{r_{if^{ap}(i)}} + \sum_l^{|\mathcal{L}(t)|} p^l_{f^{ap}(v_i)c}\Delta^l. \tag{2}$$

3) Queuing Delay: In Section III-A, we define μ_i as the service rate of the ith device. We assume that the devices in the proposed system follow the M/M/1 queuing model. Therefore, the queuing delay (δ^Q_j) at the jth device is $[1/(\sum_j^{|V_f|} x_{ij}\mu_j)]$. In this system, based on the requests from edge devices, an AA generates and distributes keys. We evaluate the queuing delay only for AA.

4) Processing Delay: Finally, we evaluate the processing delay for each of the tasks in the networks. The ith edge device requires $\Omega(E(\nabla_i))$ and $\Omega(D(\nabla_i))$ CPU cycles to encrypt and decrypt data, respectively. Therefore, the device requires the processing delay of $\Omega(E(\nabla_i))/\omega_i$ and $\Omega(D(\nabla_i))/\omega_i$ to perform the aforementioned processes. On the other hand, an AA requires Ω/ω_i unit of delay to generate secret keys

$$\delta = \left\{\left|\mathcal{V}^e\right|\right\}^{-1} \sum_{i=0}^{|\mathcal{V}^e|} \left\{t_i\delta^E + (1 - t_i)\delta^D\right\}. \tag{3}$$

In the proposed system, during the encryption phase, the ith edge device initially fetches the public key from AA. For any edge device intended to encrypt data, this process is performed once, which takes negligible unit of delay. Furthermore, the same edge device encrypts the data and transmits it to the remote server. The remote server archives the encrypted data. Thus, the total delay for the ith edge device during its encryption phase is evaluated in (2). In the proposed system, during the decryption phase, the ith edge devices initially request the AA for its secret key and the remote server for the encrypted data with negligible delay. Both the remote server and the AA simultaneously transmit the encrypted data and the secret key to the ith edge device. Based on the data and the secret key, the edge device decrypts the data. Thus, the total delay for the ith edge device during its decryption phase is evaluated in (1), shown at the bottom of the page. Additionally, the average delay for the proposed system is evaluated in (3).

D. Energy Model

Inspired by Misra and Saha [19], we define the energy consumption model for the proposed CP-ABE-based system in Society 5.0. Misra and Saha [19] defined energy consumption for local processing as $\mathcal{E}_i^{\text{loc}} = \rho(\Omega_i/\omega_i)$ for an edge device v_i, where ρ, ω_i, and Ω_i denote the energy dissipation rate of the CPU, the CPU frequency of v_i, and the CPU cycle required to process data ∇_i in v_i, respectively. The energy consumption for data transmission between v_i and v_j is computed as

$$\mathcal{E}_i^{Tx} = p_i^{Tx}\left(\Delta^{Tx}_{in} + \Delta^P_{nj}\right). \tag{4}$$

In our proposed system, the local processing includes encryption and decryption processes. Thus, we formulate the energy consumption for encryption and decryption as

$$\mathcal{E}_i^E = \rho\frac{\Omega(E(\nabla_I))}{\omega_i} \tag{5}$$

$$\delta^D = \sum_l^{|\mathcal{L}(t)|} p^l_{f^{ap}(v_i)f^f(v_i)}\Delta^l + \sum_l^{|\mathcal{L}(t)|} p^l_{f^{ap}(f^f(v_i))v_i}\Delta^l + \sum_l^{|\mathcal{L}(t)|} p^l_{f^{ap}(\mathcal{V}^c)v_i}\Delta^l + \frac{1}{\sum_j^{|V_f|} x_{ij}\mu_j} + \frac{\Omega(D(\nabla_i))}{\omega_i} + \frac{\Omega}{\omega_{f^f(v_i)}} \tag{1}$$

$$\mathcal{E}_i^D = \rho\frac{\Omega(D(\nabla_I))}{\omega_i}. \tag{6}$$

Additionally, AA requires Ω/ω_i units of time for generating the secret key. Therefore, the energy consumption for generating the secret key is evaluated as: $\mathcal{P}_i^{AA} = \rho(\Omega/\omega_i)$. Based on (4), we formulate the energy consumption for data transmission between the edge device v_i and AA $f^f(v_i)$ and between the edge device v_i and cloud c as depicted in (7) and (8), respectively

$$\mathcal{E}_{v_i,f^f(v_i)}^{Tx} = p_i^{Tx}\Big(\Delta_{v_i,f^{ap}(v_i)}^{Tx} + \Delta_{f^{ap}(v_i),f^f(v_i)}^{P}\Big) \tag{7}$$

$$\mathcal{E}_{v_i,c}^{Tx} = p_i^{Tx}\Big(\Delta_{v_i,f^{ap}(v_i)}^{Tx} + \Delta_{f^{ap}(v_i),c}^{P}\Big). \tag{8}$$

E. Packet Loss Model

We adopt the packet loss model described in [21] to evaluate the packet loss of our proposed system. In general, packet loss is evaluated as the number of successive transmissions via a link between the two devices. The number of successful transmissions varies due to the link state. The path-loss model in [21] evaluates the channel characteristics using Markov model-based path condition prediction method. This model considers two types of channel states—(1) good and (2) bad—based on the presence of noise. A channel with good state refers to a channel with less noise. On the contrary, the bad state of a channel reflects that the channel is noisy. The number of successful data transmission increases if the state of a channel transits from bad to good; otherwise, the number of successful transmissions decreases. The path-loss model in [21] designs this transition and integrates it in the Markov model to predict the status of the future packet transmissions

$$\mathcal{P}_{i,j}^{\text{rec}} = \sum_{k=1}^{\lceil \nabla_i/p \rceil} S_{i,j,k}(t). \tag{9}$$

Inspired by [21], we evaluate the packet loss $\mathcal{P}_{i,j}$ of link $l_{i,j}$ in our proposed system by formulating the number of successful packet transmissions, as shown in (9). In (9), $S_{i,j,k}(t)$ and p denote the transmission status of the kth packet over link $l_{i,j}$ at time t and packet size, respectively. The value of $S_{i,j,k}(t)$ is 1, if the kth packet is transmitted successfully; otherwise, 0. To determine $S_{i,j,k}(t)$, we design the state transition system for link $l_{i,j}$. In the system, a channel can be in either of the following two states: 1) good ($\mathcal{C}_g(t)$) and 2) bad ($\mathcal{C}_g(t)$). We also define two probability parameters—Pr_1 and Pr_2—and random variable r for designing the state transition system. If $r > Pr_1$, the link $l_{i,j}$ with ($\mathcal{C}_g(t)$) at time t remains ($\mathcal{C}_g(t+1)$) at time $t+1$; otherwise, transits to ($\mathcal{C}_b(t+1)$). Otherwise, if $r > (1 - Pr_2)$, the link $l_{i,j}$ with ($\mathcal{C}_b(t)$) at time t transits to ($\mathcal{C}_g(t+1)$) at $t+1$; otherwise, remains ($\mathcal{C}_b(t+1)$). Using this transition system, we evaluate the packet loss of our proposed system.

IV. CASE: Proposed Scheme

We propose a context-aware security scheme for preserving privacy to automatically detect the user's activity and collect the corresponding attributes that are used in the CP-ABE. The different components of CASE are illustrated as follows.

Algorithm 1: Attribute Learning

```
Input: Data ($\nabla_i$), Edge device ($v_i$)
Output: $\mathcal{A}_i$,$\nabla_i''$
$v_i$ collects $\nabla_i$ from a user;
$\nabla_i \longrightarrow \{\nabla_i', \nabla_i''\}$;
$LM(\nabla_i') \longrightarrow \mathcal{A}_i$;
return $\mathcal{A}_i$, $\nabla_i''$
```

A. Learning Attribute

CASE introduces an attribute learning scheme that automatically generates attributes from the user information. An edge device associated with a user collects his/her information, such as health parameters and activity information. We define that at any given time, the ith edge device contains ∇_i data, which is divided into ∇_i' and ∇_i''. We assume that ∇_i' contains activity information, such as walking, walking upstairs, walking downstairs, standing, and laying, and ∇_i'' contains other useful information. After collecting ∇_i, CASE extracts the attribute from ∇_i' using a pretrained learning model $LM(\cdot)$. The process for extracting attributes is discussed in Algorithm 1. CASE considers different machine learning approaches, such as DT, SVM, and NB to produce a pretrained learning model. While performing attribute learning, CASE produces two types of data: 1) attribute $\mathcal{A}_i$ and 2) raw data ∇_i''. The produced raw data (∇_i'') are encrypted using CP-ABE and $\mathcal{A}_i$.

Corollary 1: The size of ∇_i'' is less than size of ∇_i''. Thus, $S(\nabla_i'') < S(\nabla_i)$.

Proof: We prove this corollary using the approach of contradiction. Let us assume that the size of ∇_i'' is not less than the size of ∇_i''. Thus, $S(\nabla_i'') \geq S(\nabla_i)$. On the other hand, the attribute learning algorithm collects ∇_i from the ith user and divides ∇_i into ∇_i' and ∇_i''. As ∇_i'' is extracted from ∇_i, we can infer that $\nabla_i'' \subset \nabla_i$ and $\nabla_i'' \not\supset \nabla_i$, which further infers $\nabla_i'' \subseteq \nabla_i$. Therefore, as $\nabla_i'' \subseteq \nabla_i$, we can conclude $S(\nabla_i'') < S(\nabla_i)$, which contradicts the initial assumption. Finally, we conclude that the size of ∇_i'' is less than the size of ∇_i''. ■

Theorem 3: The time complexity of the aforementioned attribute learning algorithm is $\mathcal{O}(DP_i)$, where DP_i is the number of data points collected from the users using sensors.

Proof: Let us assume that the attribute learning algorithm initially collects DP_i data points from the ith user using the sensors. After collecting each of DP_i data points, the algorithm divides each of the DP_i data points into ∇_i' and ∇_i'' and extract attribute $\mathcal{A}_i$ form ∇_i'. The attribute learning algorithm returns this extracted attribute to the user. As the collection of each of the data points and dividing them into ∇_i' and ∇_i'' does not require a complicated procedure, the delay for collecting each of the DP_i data points remains constant. Therefore, the time complexity for collecting DP_i data points is $DP \times \mathcal{O}(1)$, where $\mathcal{O}(1)$ denotes the complexity of constant time process. On the other hand, extracting the attribute from ∇_i' requires a pretrained learning model $LM(\cdot)$. As the training of $LM(\cdot)$ is performed before the deployment of CASE, we consider the time to extract the attribute, which is a constant time process. Thus, for each of the DP_i data points, the time complexity for extracting an attribute is $\mathcal{O}(1)$. Moreover, the time complexity to execute the attribute learning algorithm for each of the DP_i data points is $\{\mathcal{O}(1)+\mathcal{O}(1)+\mathcal{O}(1)\}$ or $\mathcal{O}(1)$. Finally, the time complexity to execute the attribute learning algorithm for all DP_i data points is $DP_i * \mathcal{O}(1)$ or $\mathcal{O}(DP_i)$. ■

B. Preserving the Privacy of User Information

After attribute learning, CASE preserves the privacy of ∇_i'' and store it in the cloud. To preserve privacy, CASE follows the traditional CP-ABE-based security scheme [8], where we consider encryption and decryption mechanisms used in traditional CP-ABE to encrypt and decrypt data in the proposed system. The security system included in CASE comprises the following phases.

1) Setup Phase: This is the initial phase in a CP-ABE-based security scheme, which occurs during the deployment of edge devices. An edge device v_i shares its security information in the setup phase, such as its ID with the associated AA. Based on the received security information, the associated AA produces a public key (PCK_i) and a master key (MK_i). Thereafter, the associated AA replies PCK_i to the edge device.

2) Key Generation Phase: Before the data transmission in CP-ABE-based security, the edge device v_i collects the attributes from the user and transmits these to the associated AA. The associated AA generates the security key (SK_i) using the master key MK_i, produced in the setup phase. Additionally, the same AA archives the security key and does not transmit it to the edge device v_i. It is noteworthy to mention that the system does not collect the attribute from the user manually in CASE. Instead, the system learns the attribute by the information accumulated from the sensors attached to the human body. We discussed the procedure to learn the attribute in Section IV-A. The edge device transmits these learned attributes to the associated AA for generating the security key.

3) Encryption Phase: This phase helps in selecting a policy string Pol_i based on the users' requirements. Policy string in CP-ABE is Boolean information among the attributes, which enforces a restriction on data decryption. For better understanding, we consider two policy strings: 1) *walking and running and (doctor or physical instructor)* and 2) *laying and (doctor)*. The first policy string denotes that users with either doctor or physical instructor attributes can access data with walking and running attributes. On the other hand, the later policy string specifies that a user having a doctor attribute can access the data with a laying attribute. Using the public key and the specified policy string, this phase encrypts the data and transmits the encrypted data to store it.

4) Decryption Phase: In this phase of the CP-ABE security system, a user tries to access the encrypted data in the cloud. To access the data, the user transmits his/her attribute to the AA and simultaneously sends a request to the cloud to access the data. Based on the received attribute, the AA identifies the secret key SK_i and shares it with the user. After receiving the secret key from the AA and data from the cloud, the user decrypts the encrypted data.

C. Theoretical Analysis

We evaluate the performance of CASE in comparison with two existing approaches CSCPABE [6] and PBCPABE [7]. These approaches focus on providing solutions that reduce the post-encryption data size in a CP-ABE-based security system. However, these approaches lack the perspective to automatically generate user attributes from the user information collected from the sensors.

Theorem 4: CASE reduces the post-encryption data size as compared to the existing approaches e.g., [6] and [7].

Proof: Let us assume that both the existing approaches [6], [7] and CASE encrypt the data ∇_i using the encryption function $E(\cdot)$. The existing approaches encrypt ∇_i using the function $E(\cdot)$ and produce cipher text $E(\nabla_i)$. The size of the cipher text is $S(E(\nabla_i))$ On the other hand, CASE divides ∇_i into ∇_i' and ∇_i''. Furthermore, ∇_i' is used to generate attributes and ∇_i'' is encrypted using the generated attributes and $E(\cdot)$. After encryption, CASE produces the cipher text $E(\nabla_i'')$. As motioned in Corollary 1 that $S(\nabla_i'') < S(\nabla_i)$, we can state that $S(E(\nabla_i''))$ is also less than $S(E(\nabla_i))$. Thus, Theorem 4 reports true. ■

Theorem 4 theoretically proves that the size of the post-encryption data produced by CASE is reduced in comparison with the existing approaches [6], [7]. For enabling CASE, we proposed a CP-ABE-based system in Society 5.0 and define delay, energy consumption, and packet loss model, which is dependent on the post-encryption data size. Thus, we theoretically state that CASE reduces the average delay, energy consumption, and packet loss compared to the existing approaches.

V. Performance Evaluation

We used a pretrained model for extracting attributes from the users' information collected from the sensors. The model is trained in a computation-intensive machine using the UCI activity data set [22]. This data set contains 561 features and 7351 records. Additionally, this data set records human activities, such as walking, walking upstairs, walking downstairs, standing, and laying. One-third of the same data set is used for predicting the attribute, and the remaining data set is used for encryption. We opt for a Java-based CP-ABE system [23] to encrypt and decrypt the data.

A. Experimental Setup

We use different parameters for evaluating CASE, which are depicted in Table I. We evaluate the proposed system with 1000, 2000, and 3000 edge devices, 1500 access points, 20 fog devices, and a cloud platform. These devices are deployed randomly over 1×1 km^2 area. The edge devices transmit data with 100, 200, and 300 MB size to the cloud server. Furthermore, they transmit the data with 2.2-W transmission power [19] over the channel having -100 dB noise [19] and 5 Mb/s bandwidth [19]. Such channel changes from good to bad state with probability 0.02777 and *vice versa* with probability 0.25 [21]. We consider the communication range of an edge device and an access point to be 100 and 350 m, respectively, [19]. We design the simulation based on the network model mentioned in Section III.

1) Performance Metrics: We evaluate the performance of CASE using different metrics—average network delay, average energy consumption, and average packet loss. These

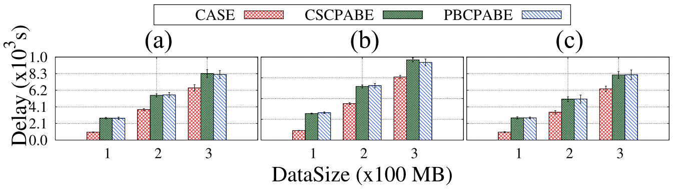

Fig. 3. Average delay. (a) Number of edge devices = 1000. (b) Number of edge devices = 2000. (c) Number of edge devices = 3000.

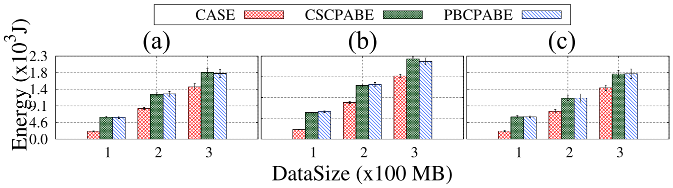

Fig. 4. Average energy consumption. (a) Number of edge devices = 1000. (b) Number of edge devices = 2000. (c) Number of edge devices = 3000.

TABLE I
EXPERIMENTAL PARAMETER

Parameters	**Value**
Noise (dB) [19]	-100
Bandwidth (MHz) [19]	5
Pr_1 [21]	0.027
Pr_2 [21]	0.25

TABLE II
REDUCTION IN THE NETWORK PARAMETERS

Parameter	**CSCPABE**	**PBCPABE**
Network Delay	32%	33%
Energy Consumption	33%	35%
Packet Loss	31%	36%

TABLE III
NUMERICAL ANALYSIS OF LEARNING PARAMETERS

Parameter	**NB**	**SVM**	**DT**
Prediction Time (s)	31	603	2.16
Clock cycle ($E + 03$)	62.4	1200	4.31
Accuracy (%)	64.4	99.3	99.7

performance metrics are evaluated with a varying number of edge devices and data sizes, as depicted in Figs. 3–5. We compare CASE with the existing approaches—CSCPABE [6] and PBCPABE [7], which are discussed in Sections II and IV-C. By evaluating the impact of the varying edge nodes and data sizes on these performance metrics, we determine the network lifetime using CASE. On the other hand, the scalability of CASE is inferred by studying the impact of varying edge nodes and data size on these metrics. To evaluate the execution feasibility of the attribute learning scheme in the edge devices, we analyze the learning metrics, including average prediction time, average clock cycle requirement, and average accuracy, as illustrated in Fig. 6. We analyze these learning parameters over NB, SVM, and ST-based pretrained models by predicting data sets with 2000, 4000, and 6000 entries. Based on this analysis, we evaluate the performance of these pretrained models in edge devices.

B. Results and Discussions

We elaborately discuss the analysis of the results that we obtained from our simulation. We also discuss the performance of CASE in contrast to the existing approaches—CSCPABE [6] and PBCPABE [7].

1) Analysis on Network Metrics: This article considers average network delay, average energy consumption, and average packet loss as network metrics for evaluating the performance of CASE. In Figs. 3–5, we depict the obtained results of these metrics over a varying number of edge nodes and the data size. We also tabulate the numerical results of evaluation in Table II. These results report that CASE reduces the average network delay, the average energy consumption, and the average packet loss, in contrast to the existing approaches—CSCPABE [6] and PBCPABE [7]. In Section IV-C, we theoretically prove that CASE reduces the size of the post encrypted information with respect to the same approaches by using an attribute learning scheme. Furthermore, in that section, we prove that CASE reduces these network metrics' values due to the reduction in the post-encrypted information size. On the other hand, in Figs. 3–5, we consider the presence of 1000, 2000, and 3000 edge nodes in the network. In each of these cases, we observe that CASE reduces average delay, average energy consumption, and average packet loss compared to CSCPABE and PBCPABE. Thus, it seems that the results depicted in Figs. 3–5 and in Table II justify this theoretical analysis.

The network lifetime of any system inversely depends on the average energy consumption. Moreover, network lifetime measures the operating time of a network. The increase in average energy consumption reports that the network devices consume their energy resources at an increased rate. Due to such a consumption rate, the network may not provide services to the users, and the network lifetime reduces. On the contrary, we observe in Fig. 4 and Table II that CASE reduces the energy consumption by 33% and 35% in contrast to the existing approaches. Therefore, CASE increases the network lifetime.

2) Analysis of Learning Parameters: CASE allows the edge devices to perform attribute learning using a pretrained learning model. It extracts the attribute by predicting user activity. The pretrained model is trained on a computationally rich machine. Thus, we analyze the learning parameters of activity prediction in CASE. We consider average prediction time, average clock cycle requirement, and average accuracy as the learning parameter. We depict the evaluated results in Fig. 6 and tabulate the numerical analysis in Table III. In these analytical results, we observe that the NB-based pretrained model performs worse than both SVM and DT-based pretrained models, in terms of prediction accuracy. On the other hand, the analysis reports that both the SVM and DT-based pretrained models provide high prediction accuracy. However, in terms of the prediction time and average clock cycle requirements, the SVM-based pretrained model shows slower execution than both the NB and DT-based pretrained models. Thus, a detailed analysis shows that both SVM and DT-based

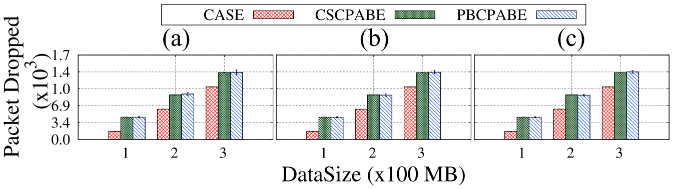

Fig. 5. Average packet loss. (a) Number of edge devices = 1000. (b) Number of edge devices = 2000. (c) Number of edge devices = 3000.

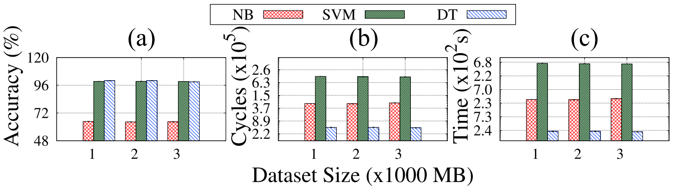

Fig. 6. Learning parameters during attribute prediction. (a) Average accuracy. (b) Average clock cycles. (c) Average time.

pretrained models can be used in the edge devices, but the DT-based pretrained model performs more efficiently than the SVM-based pretrained model.

3) Analysis of the Scalability of CASE: We analyze the scalability of CASE by observing the impact of a varying number of edge devices and data size on the network metrics. The average network delay, average energy consumption, and average packet loss are intertwined network metrics. We obtain the impact of varying edge devices and data size on average energy consumption and average packet loss by observing the impact of the same on average delay. Fig. 3 shows the variations in average network delay over varying edge nodes and data size. This result reports a linear increase in average network delay in contrast to the linear increase in edge nodes and data size. Thus, we state that CASE is scalable.

VI. Conclusion

This article introduced a CASE for CP-ABE-based security systems that uses the facilities provided by edge intelligence in IoT-enabled Society 5.0. For implementing CASE, this work also designed a CP-ABE-based security system in the context of SHS. In the system, CASE collects user information from edge device using IoT sensors. This information is further divided into two sets, one of which automatically extracts the users' contextual information and generates attributes. Considering SHS, the patient's information is divided into two sets, one of which is activity information. In this context, CASE analyzes the user's activity and generates attributes in the edge devices. Additionally, this attribute is further used to encrypt the user's remaining set of information. Using an attribute learning scheme, CASE reduced the post-encryption data size. Through extensive experiments, we observed that CASE performed better than the existing CP-ABE-based security schemes—(CSCPABE [6] and PBCPABE [7]). This work can be extended in the future to automate the generation of policy-string in CP-ABE-based security systems in IoT-enabled Society 5.0. Using such automation may help in eliminating unauthenticated information access due to the user's incorrect selection policy string.

References

[1] M. Fukuyama, "Society 5.0: Aiming for a new human-centered society," *Japan Spotlight*, vol. 27, pp. 47–50, Jul./Aug. 2018.

[2] P. Huang, L. Guo, M. Li, and Y. Fang, "Practical privacy-preserving ECG-based authentication for IoT-based healthcare," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9200–9210, Oct. 2019.

[3] S. Misra, A. Mukherjee, A. Roy, N. Saurabh, Y. Rahulamathavan, and M. Rajarajan, "Blockchain at the edge: Performance of resource-constrained IoT networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 1, pp. 174–183, Jan. 2021.

[4] S. Kavitha, P. J. A. Alphonse, and Y. V. Reddy, "An improved authentication and security on efficient generalized group key agreement using hyper elliptic curve based public key cryptography for IoT health care system," *J. Med. Syst.*, vol. 43, p. 260, Jul. 2019.

[5] D. D. Dharamadhikari and S. C. Tamane, "Secure cloud-based E-healthcare system using ciphertext-policy identity-based encryption (CP-IBE)," in *Smart Trends in Computing and Communications. Smart Innovation, Systems and Technologies*, Y.-D. Zhang, J. K. Mandal, C. So-In, and N. V. Thakur, Eds. Singapore: Springer, 2020, pp. 199–209.

[6] W. Susilo, G. Yang, F. Guo, and Q. Huang, "Constant-size ciphertexts in threshold attribute-based encryption without dummy attributes," *Inf. Sci.*, vol. 429, pp. 349–360, Mar. 2018.

[7] V. Odelu, A. K. Das, Y. S. Rao, S. Kumari, M. K. Khan, and K.-K. R. Choo, "Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment," *Comput. Stand. Interfaces*, vol. 54, pp. 3–9, Nov. 2017.

[8] M. B. Taha, C. Talhi, and H. Ould-Slimanec, "A cluster of CP-ABE microservices for VANET," *Procedia Comput. Sci.*, vol. 155, pp. 441–448, Aug. 2019.

[9] C. Jin, X. Feng, and Q. Shen, "Fully secure hidden ciphertext policy attribute-based encryption with short ciphertext size," in *Proc. 6th Int. Conf. Commun. Netw. Security*, 2016, pp. 91–98.

[10] S. Rana and D. Mishra, "Efficient and secure attribute based access control architecture for smart healthcare," *J. Med. Syst.*, vol. 44, no. 5, pp. 1–11, 2020.

[11] K. Sowjanya, M. Dasgupta, and S. Ray, "A lightweight key management scheme for key-escrow-free ECC-based CP-ABE for IoT healthcare systems," *J. Syst. Archit.*, vol. 117, Aug. 2021, Art. no. 102108.

[12] F. Khan, S. Khan, S. Tahir, J. Ahmad, H. Tahir, and S. A. Shah, "Granular data access control with a patient-centric policy update for healthcare," *Sensors*, vol. 21, no. 10, p. 3556, 2021.

[13] N. Saravanan and A. Umamakeswari, "HAP-CP-ABE based encryption technique with hashed access policy based authentication scheme for privacy preserving of PHR," *Microprocess. Microsyst.*, vol. 80, Feb. 2021, Art. no. 103540.

[14] S. Khalifa, G. Lan, M. Hassan, A. Seneviratne, and S. K. Das, "HARKE: Human activity recognition from kinetic energy harvesting data in wearable devices," *IEEE Trans. Mobile Comput.*, vol. 17, no. 6, pp. 1353–1368, Jun. 2018.

[15] P. Bharti, D. De, S. Chellappan, and S. K. Das, "HuMAn: Complex activity recognition with multi-modal multi-positional body sensing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 4, pp. 857–870, Apr. 2019.

[16] E. Casella, M. Ortolani, S. Silvestri, and S. K. Das, "Hierarchical syntactic models for human activity recognition through mobility traces," *Pers. Ubiquitous Comput.*, vol. 24, no. 3, pp. 451–464, 2020.

[17] R. Janarthanan, S. Doss, and S. Baskar, "Optimized unsupervised deep learning assisted reconstructed coder in the on-nodule wearable sensor for human activity recognition," *Measurement*, vol. 164, Nov. 2020, Art. no. 108050.

[18] N. Dua, S. N. Singh, and V. B. Semwal, "Multi-input CNN-GRU based human activity recognition using wearable sensors," *Computing*, vol. 103, pp. 1461–1478, Mar. 2021.

[19] S. Misra and N. Saha, "Detour: Dynamic task offloading in software-defined fog for IoT applications," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1159–1166, May 2019.

[20] A. S. Sohal, R. Sandhu, S. K. Sood, and V. Chang, "A cybersecurity framework to identify malicious edge device in fog computing and cloud-of-things environments," *Comput. Security*, vol. 74, pp. 340–354, May 2018.

[21] S. Jelassi and G. Rubino, "A perception-oriented Markov model of loss incidents observed over VoIP networks," *Comput. Comm.*, vol. 128, pp. 80–94, Sep. 2018.

[22] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. 21st ESANN Comput. Intell. Mach. Learn.*, 2013, pp. 437–442.

[23] J. Wang. (2012). *Java Realization for Ciphertext-Policy Attribute-Based Encryption*. [Online]. Available: https://github.com/junwei-wang/cpabe/