World Academy of Science, Engineering and Technology
International Journal of Industrial and Manufacturing Engineering
Vol:17, No:7, 2023

IoT Device Cost Effective Storage Architecture and Real-Time Data Analysis/Data Privacy Framework

Femi Elegbeleye, Seani Rananga

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

***Abstract*—**This paper focused on cost effective storage architecture using fog and cloud data storage gateway, and presented the design of the framework for the data privacy model and data analytics framework on a real-time analysis when using machine learning method. The paper began with the system analysis, system architecture and its component design, as well as the overall system operations. Several results obtained from this study on data privacy models show that when two or more data privacy models are integrated via a fog storage gateway, we often have more secure data. Our main focus in the study is to design a framework for the data privacy model, data storage, and real-time analytics. This paper also shows the major system components and their framework specification. And lastly, the overall research system architecture was shown, including its structure, and its interrelationships.

***Keywords*—**IoT, fog storage, cloud storage, data analysis, data privacy.

I. Introduction

IoT is the revolutionary idea that connects computing devices and enables the transfer of data over the network without human interaction. As communication between day-to-day devices becomes autonomous, IoT makes everything from household appliances to automobiles, smart cities, and device interactions very easy. However, in IoT, high volumes of data are generated on an hourly or daily basis from the IoT devices and they are vulnerable to some threats from malicious activities. Moreover, most of the data generated from these devices, such as healthcare applications, contain sensitive information or personally identifiable information about patients which are meant for healthcare practitioners and the patients only. Therefore, proper measures must be put in place to protect the data and this forms one of the motivations for this research. This stems from the high value placed on data globally as it is considered one of the most valuable assets of organizations that are critical for informed decisions making and productivity. Moreover, this paper also proposes a less costly storage platform for the generated high-volume data from IoT devices to help reduce the cost of access and efficient processing for real-time decision making. In general, the generated data must be effectively protected from the perspective of data privacy, cost-effective storage, and real-time analysis for effective decision making aimed at promptly solving some of the noticeable challenges. This paper, therefore, presented the proposed system functions, the framework design using the chosen data privacy model and the data analytic model chosen when being applied on a real-time data analysis which aids an effective decision-making process.

Femi Abiodun Elegbeleye is with Department of Information Techology Systems, Walter Sisulu University, Komani, South Africa (e,mail: felegbeleye@wsu.ac.za).

Seani Rananga is with Department of Computer Science, University of Pretoria, Lynwood Hatfield, South Africa (e-mail: Seani.rananga@up.ac.za).

II. System Analysis

System analysis is an important activity that is conducted during software development and is done before the system is designed and implemented. As a key problem-solving technique, it involves having a good understanding of the components that make up the system, how they operate and the relationships among them. To design an effective system model, there is a need to gather and analyze the important system requirements via necessities elicitation and analysis which are performed from the point-of-view of consistency, validity, and feasibility. Performing these activities helps with the identification of the key requirements and the constraints or quality imposed on their operations.

This section, therefore, provides the proposed system functions which are critical to meeting the overall goal of this research and the designed model quality. The elicitation technique applied in this research is based on a comprehensive literature study that centered on understanding and analyzing existing data privacy models, cloud storage architectures and data analytic techniques. Table I presents the basic functional and non-functional requirements of the system that would be achieved.

TABLE I
Functional and Non-Functional Requirements of Proposed System

Req. Id	Functional Requirements
R1	The system senses and monitors data from the IoT device
R2	The system filters all data from the IoT device
R3	The system processes/manages all data from the IoT device
R4	The system encrypts all data to make sure data is secure
R5	The system applies a data privacy model to protect data
R6	The system applies data anonymization before data is stored
R6	The system utilizes the ML algorithm to predict its functions
R7	The system utilizes the fog gateways system in storing all data
	Non-Functional Requirements
R8	The system accounts for every loss of data
R9	The system stores a large volume of data

A. Use Case Model

Based on the system requirements in Table II, Fig. 1 shows the use-case model of the proposed system showing the actors, use-cases, and their interaction with the system. The objective is to enhance the understanding of the overall system function.

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

TABLE II
USE CASE DESCRIPTION

IoT Device	IoT Gateway	Fog Server	Cloud
Monitor, Sense data	Device connectivity, security data filtering data processing	Data processing, Data privacy, Data analytic - ML Cleaning/Filtering Data storage	Data processing, Data privacy, Data analytic - ML Cleaning/Filtering Data storage

As shown in Fig. 1, the proposed system is centered on IoT data privacy, storage, and analytics to effectively protect the generated data, cost-effectively store them, and analyze the data on a real-time basis for informed decision making. Furthermore, the use-case descriptions of each core functionalities shown in Fig. 1 is presented in Tables III-VI.

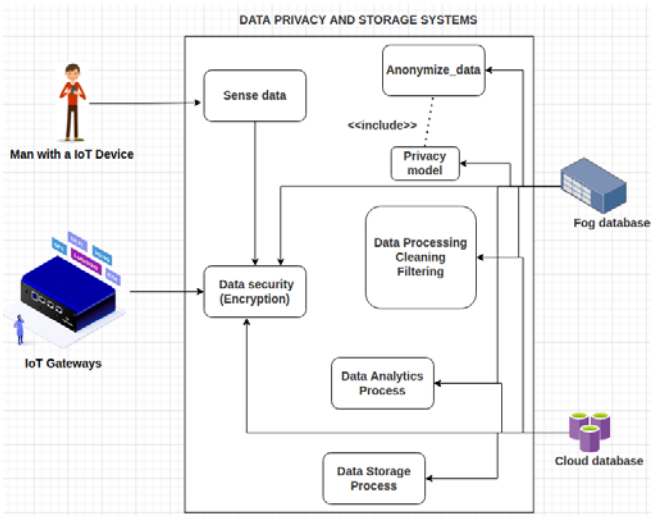

Fig. 1 System use-case diagram

TABLE III
ACTOR ROLES

Use Case Name	Sense Data
Actor	IoT Device
Brief Description	System users /patients and healthcare staff IoT devices process the data Perform operations on a real-time interval IoT device sends out data into the fog gateway/cloud IoT devices get instant feedback from either fog getaway/cloud Get the actual real-time action Get the location of the action Decide what to do with the violator
Precondition	IoT device shall have a unique id (for unique identification) IoT devices must switch be on IoT devices must be configured or set to send and receive all data IoT device must be configured to get alert responses on a real-time interval Provide end-to-end encryption for all communication between IoT devices, Fog gateway or Cloud
Post Condition	IoT device sense patients/healthcare staff unique id IoT devise sense intruder once all precondition is violated and report IoT device sense validates if all condition is meant Send an alert message to either patient/healthcare staff

TABLE IV
DATA ANONYMIZATION

Use Case Name	Anonymize data
Actor	Fog
Brief Description	Import data and configure the system Eliminating personally identifiable information Apply generalization and suppression Filter the dataset and choose transformations model Choose data privacy model Organize and secure dataset
Precondition	The dataset must properly be filtered Choose the right data privacy model (KA and DP)
Post Condition	Reject the final output if all conditions are not fulfilled Accept the final output if all conditions are meant

TABLE V
DATA ANALYTICS

Use Case Name	Data_ Analytics
Actor	Fog, Cloud
Brief Description	Choosing the dataset Trained and rest dataset Additional data processing Additional data cleaning Data analysis Communication
Precondition	Variable selection Data visualization
Post Condition	Reject the final output if all conditions are not fulfilled.

TABLE VI
DATA STORAGE

Use Case Name	Data storage
Actor	Fog and Cloud
Brief Description	• Set time-sensitive operation • Do information profiling • Checked for service performance • Service coordination • Stored data.
Precondition	• Accept only clean data and store it
Post Condition	• Reject the final output if all conditions are not fulfilled. • Accept the final output if all conditions are meant

B. System Sequence Diagram

The diagram representation shown in Fig. 2 illustrates the sequence of activities of the proposed system. It typically depicts the interactions between the IoT devices from the various users engaged with the IoT gateway before processing the data generated from the IoT device. The next step is to anonymize the sensitive data in the fog gateway after which the data are being processed further to better ensure data integrity. In the fog gateway, we adopt a dual step data privacy model on the received sensor data; this helps to better protect the privacy of the data from an unauthorized person for them not to be able to gain any useful information or knowledge from the data, and the process is done on a real-time basis. The privacy model further helps whosoever is assigned to manage such data properly by protecting all data which also enhances an effective dynamic decision process. Because the fog gateway processes are closer to the IoT device data, then data can also be stored for a quicker reply to all queries. And, the last step is the cloud database system where data can also process and stored permanently.

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

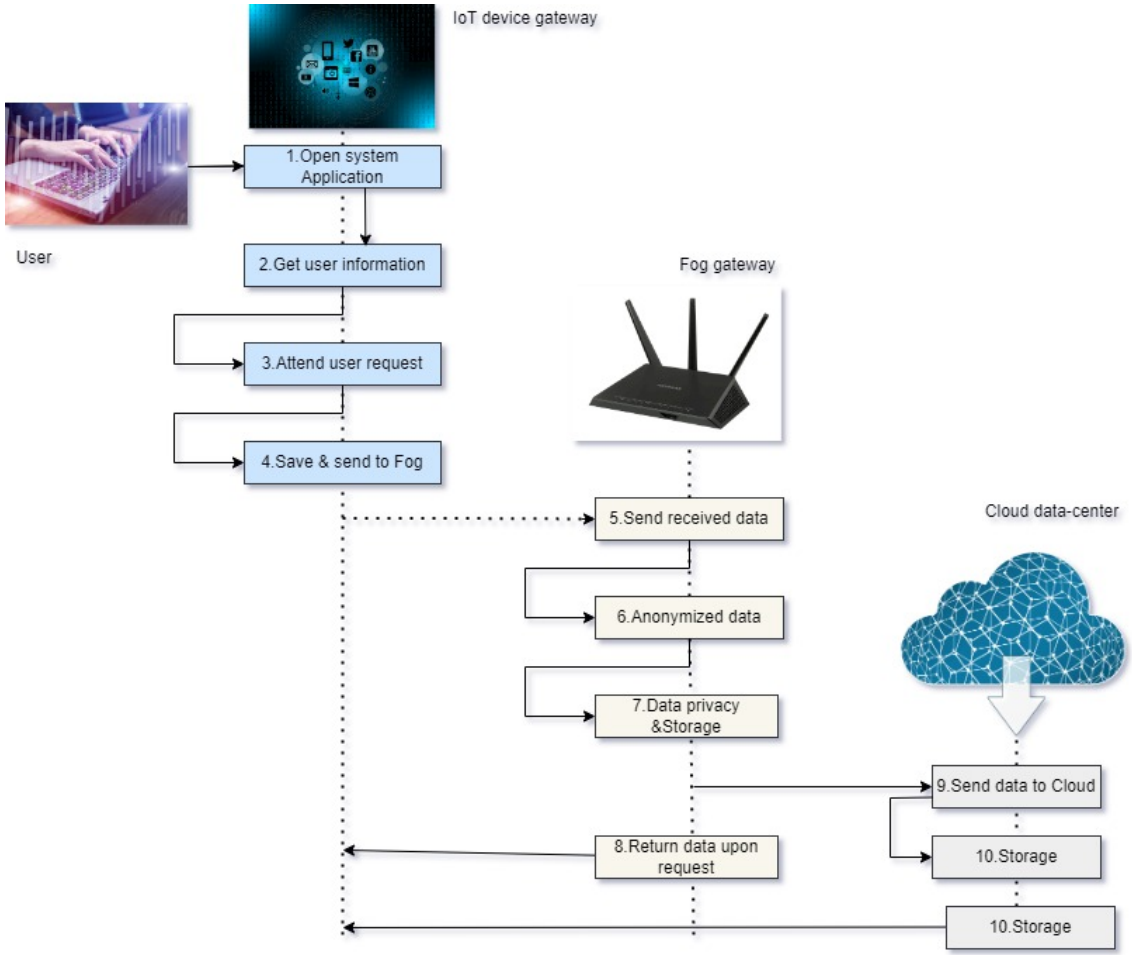

Fig. 2 System sequence diagram

This section presented the proposed cost-effective storage architecture that achieves the privacy and real-time data analytics of the high volume of it generated from every IoT device. The IoT Fog-based architecture is shown in Fig. 3 and consists of different layers: The Connected "Things", IoT Gateway, Utility Center, Fog Server, and the Cloud. After an in-depth literature survey, this research proposed the integration of Fog Server and the Cloud as the cost-effective storage for IoT generated data where data privacy can be ensured as well as real-time analytics.

The components in each of the layers are discussed. For clarity and comprehension, this research used the healthcare sector as an illustration.

1) Layer 3: Connecting "Things" and Devices

Connected Things or IoT device (mobile device) serves as a human and system interface that gathers or generates data using the embedded sensors. The system can be used to detect and predict any usual and unusual situation in the environment (i.e., human, animals, physical environment, etc.) in which they are installed. For instance, in the healthcare perspective, the devices could be used to monitor situations such as high/low blood sugar, cardiac arrest or blood pressure, etc. In this case, the sensors capture the data and send it to the fog servers or the cloud via the gateway.

The connected "Things" lie at the lowest layer of the fog-based computer setting where end-users can access data from the fog nodes or send localized data to the fog. However, if data are not available in the fog or the fog is unavailable, the cloud storage in the cloud server may be accessed directly. In some cases, and the absence of an IoT gateway, the sensors can be equipped with the task of data pre-processing, filtering, cleaning, and encrypting the data before sending it into fog sites or cloud sites for further processing.

Utility Center: This module is responsible for the consumption of the data stored in the fog for decision making. It could be organizations like hospitals, weather stations, etc. that are permitted to have access to the stored data. For instance, in the case of the healthcare sector or hospital, the patient must register in a particular hospital. The hospital will then access the patient's data and then take immediate necessary action. Also, the data are made private (anonymized) and secure to avoid unauthorized access. The patient would not be required to share the data with the hospital physically and the hospital would access only the essential data required during a medical check-up.

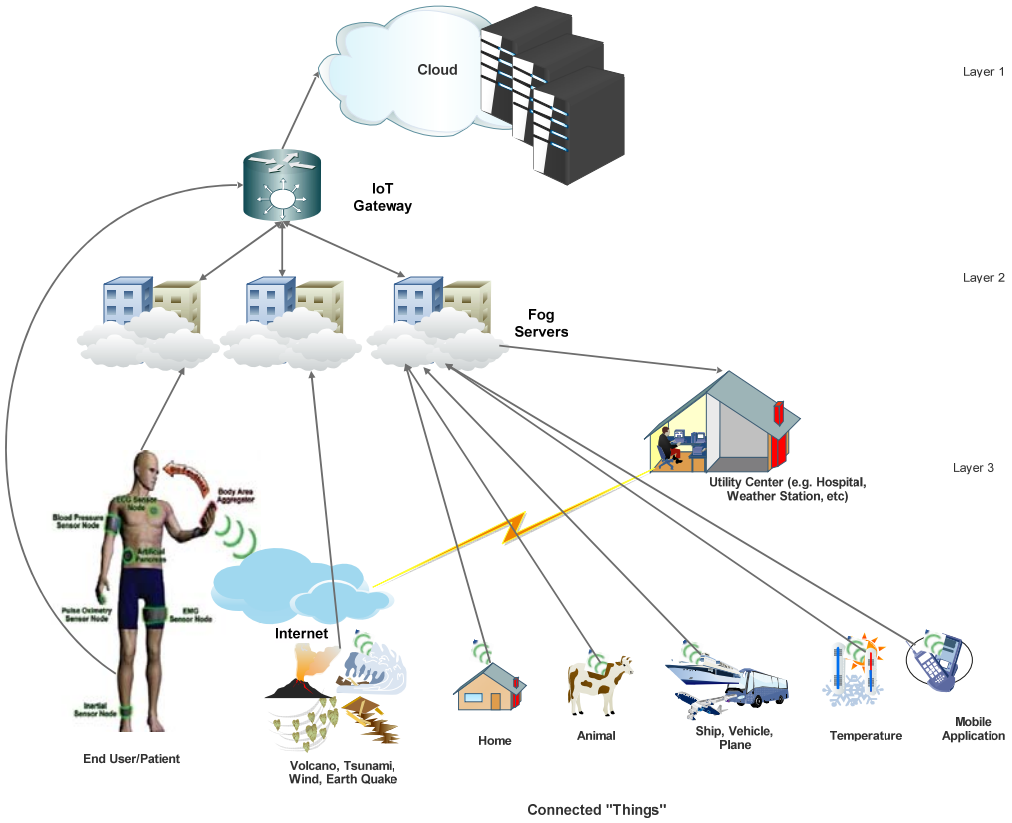

Fig. 3 System architecture diagram

2) Layer 2: Fog Servers

This is the second layer of the architecture between the connected devices and the cloud. This is the most important layer of this architecture in terms of computing. The fog layer consists of fog nodes which could be routers, base stations, proxy servers, and so on. The tasks of the nodes are to receive the sent data from the connected devices, process them, and temporarily store the data for onward use at the utility center or end users. It then sends it to the cloud for large-scale storage. Also, the fog nodes receive data that do not meet the requirements of the end-users from the cloud which are then processed for user consumption accessed via internet communication. In each fog node, several operations can be performed on the data in real-time such as real-time data processing, data analytics, ML, data privacy, data caching, computation offloading, etc., to support effective and dynamic decision making.

3) Layer 1: Cloud Servers

This is the highest layer of the computational architecture. It has a series of centralized data centers equipped with the capability to store all data received from the fog servers. Its capacity is so huge that it can store a large volume of generated data. Moreover, due to this capacity, there is always cases of high network congestion as well as high latency in QoS delivery. This is the reason why fog nodes are introduced to avoid the problems associated with the cloud, especially where time efficiency is of the essence. Moreover, the cloud layer performs essential tasks just like the fog servers, but its capacity and processes take a much longer time when compared with using fog. In the context of this research, the cloud site was used for data storage only while all the other processes were carried out in the fog servers.

IoT Gateway: This is a hardware device with application software that accomplishes essential tasks. It is used to enable IoT communication which is either device-to-device or device-to-cloud. That is, the gateway facilitates different data sources and destinations. Some of the tasks performed by the gateway include data caching, data pre-processing, data filtering, data cleaning, optimization, network features, security, and device management.

C. Fog and Cloud-Based Architectural Design

In recent years, cloud computing has gained momentum among individuals and organizations in terms of its computing, data storage and network management roles [1]. These functions are performed in centralized data centers, thereby providing cost-effective services, increased internet access, productivity, performance, security and reliability [2], [3]. However, centralized data centers often fail to meet the requirements of billions of geographically distributed IoT devices [4]. That is, they fail to offer real-time services, cause high latency in the delivered QoS, high network congestion, etc. To address these demands, the “fog computing” concept was introduced by Cisco [5] as an extension of cloud-based facilities which are closer to the IoT nodes (devices). Fog

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

nodes interface the end-users and the cloud data centers and are decentralized in nature. Due to the several advantages noted with fog data storage such as low latency, support of mobility, support of real-time services, low power consumption, geographical distribution, cost-effectiveness, reliability, less congestion of network and processing a high number of nodes [6], organizations that wish to safe cost can leverage on this to reduce the huge cost used when using cloud data storage. due to the advantages they offer [6], [7].

These are the motivations for adopting a fog-based architecture for IoT generated data. In the context of this research, a typical fog computing architecture is shown in Fig. 4. In the second layer, which is the fog node, several tasks or operations can be performed on the data collected from the end-users or IoT devices. However, based on the objectives outlined in this research, we focused only on data privacy and real-time data analytics.

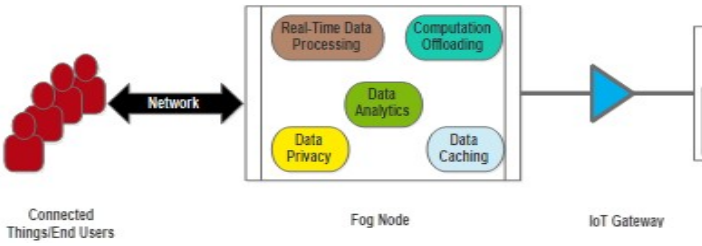

Fig. 4 Fog-cloud architecture

K-Anonymity Algorithm

Input: k-anonymous database D, centroids $C = c_1, \ldots, c_m$, partitions $P = P_1, \ldots, P_m$, counts $K = k_1, \ldots, k_m$, timestamp t, operation σ.
Output: k-anonymous database D_t, updated centroids C_t, and counts K_t,

if $\sigma = add((x, t), D)$ **then**
$b = argmin_i\ d(x, c_i)$ (Add x to group C_b)
$k_b = k_b + 1$
if $k_b + 1 = 2k$ **then**
$(P_b, P_m + 1) =$ Apply MDAV to the points in P_b
$C = C \setminus c_b$
$C = C \cup \{c_b, C_{m+1}\}$
$P_b = \emptyset$
end
end
if $\sigma = remove((x, t), D)$ **then**
$b = argmin_i\ d(x, c_i)$ (assign b to buffer of removals R_b)
$k_b = k_b - 1$
if $k_b = 0$ **then**
$C = C \setminus c_b$
$R_b = \emptyset$
end
end
return (D_t, C_t, K_t, P_t)

Fig. 5 K-Anonymity algorithm [12]

Fig. 4 shows the relationship between IoT devices, fog nodes and cloud data centers. This has various opportunities as seen from previous studies of [8], [9]. Most organizations have realized the advantages of using fog and cloud in their daily operations.

1) Data Privacy Model

This subsection presented the proposed approach to make the sensitive data generated by IoT devices private and confidential in the fog servers or nodes. To achieve data privacy, data anonymization is the key. Several data privacy models exist each having its strengths and weaknesses [10], [11]. We have no plans to develop a brand-new data privacy model. Four of the currently used data privacy models were used in this work. Two out of the four data privacy models will be chosen based on how well they performed. With the analysis conducted, the DP and KA data privacy models were chosen as the most effective of the four models considered. It is believed that when DP and KA are combined, the privacy of generated data is well protected against all forms of attacks and from unauthorized access. The detail of the process involved is explained using the KA and DP algorithms that were merged as shown in Figs. 5 and 6.

Differential Privacy Algorithm

Input: private database D, an adaptively chosen stream of sensitivity L queries $f_1, \ldots$, a threshold T, and a cutoff point c.
Output: stream of answers $a_1, \ldots$

Sparse$(D, \{f_1\}, T, c, \epsilon, \delta)$
If $\delta = 0$, **then**
let $\sigma = \frac{2c}{\epsilon}$.
else
$\sigma = \sqrt{\frac{32c \ln \frac{1}{\delta}}{\epsilon}}$
end if
Let $\hat{T}_0 = T + Lap(\sigma)$
Let count = 0
For each query i do
Let $v_i = Lap(2\sigma)$
If $f_i(D) + v_i \geq \hat{T}_{count}$ **then**
Output $a_i = T$
Let count = count + 1
Let $\hat{T}_{count} = T + Lap(\sigma)$
else
Output $a_i = \perp$
end if
if count $\geq c$ **then**
Halt.
end if
end for

Fig. 6 Differential Privacy algorithm [13]

2) The K-Anonymity

From the four data privacy models that were analyzed in this paper, the results obtained showed that none of the four data privacy models' analyses is all-sufficient on its own to completely protect the dataset. Therefore, the need to combine two of the models arises which we initially proposed to do after reading the literature and the shortcomings associated with each of them. Accordingly, KA was chosen after many considerations as identified in the literature. Fig. 5 shows the various steps followed when applying the KA data privacy model to protect the dataset. KA usually protects the data and ensures that it can be able to be equated to k > 1. Though KA cannot protect against background knowledge attacks, this prompted us to combine it with DP to adequately address that challenge.

Table VII shows KA data privacy notations, it also shows what each symbol stands for as represented in Fig. 5. The algorithm in Fig. 5 gives the concise steps followed when applying KA on the dataset. To further illustrate this, Table VIII contains the raw dataset used to show how the KA model works. For instance, when considering the age attribute in Table VIII that is the quasi-identifier, generalization and suppression method was been used by setting the range of age, which then help to further confuse the adversary, and also, by removing some digits from the mode of identification we also

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

deleted the gender from some of the columns in the table as shown in Table IX, the Id and gender of the users can be generalized by first removing some of the user's gender and Id digits from the table.

TABLE VII
KA NOTATIONS

Notation	Meaning
KA	k-anonymity
D	database
T	timestamp
K	counts
Σ	operation
P	partition
D_t	output
C	centroids
C_t	updated centroids

TABLE VIII
KA DATA

Quasi-Identifier				Sensitive Identifier
Age	Gender	Zip code	Id	Disease
34	Male	45673	110003442	COVID-19
69	Male	64747	110003441	Ebola
76	Female	23456	776564441	HIV
45	Female	35678	776564442	Flu
76	Female	23498	847474742	Diabetes
76	Male	65489	847474743	Cholera

TABLE IX
KA-ANONYMIZED DATA

Quasi-Identifier				Sensitive Identifier
Age	Gender	Zip code	Id	Disease
(34-25)	Male	45673	1100034**	COVID-19
(69-35)	Male	64747	1100034**	Ebola
(76-30)	Female	23456	7765644**	HIV
(45-20)	Female	35678	7765644**	Flu
76	*	23498	8474747**	Diabetes
76	*	65489	8474747**	Cholera

D. Differential Privacy (DP)

DP is also considered to be the most suitable data privacy model when compared with the four other data privacy models analyzed. DP data privacy model usually takes the ML model approach using the Gaussian naïve Bayes classifier method which will be trained and add Laplacian noise concerning the ε while computing Gaussian mean and variance. In this paper default ε =1 is used [13]. Table X shows the various notations used in the DP data privacy algorithm.

With the application of the DP data privacy model, we can solve most of the other limitations that were noticed in the other data privacy model. But DP itself is not sufficient to solve every problem to which the stored data are exposed because DP privacy has its drawbacks also, so it was suggested we combine K-Anonymity with differential privacy which would further enhance the level of protection on the data. The algorithm presented in Fig. 6 shows how the DP model can be achieved. Table XI represents the anonymized dataset when applying the DP privacy model. These results, as seen in Table XI, show that DP can better protect the dataset, and this further corroborates what was stated in the literature. This method requires shuffling of the dataset that is, hiding the user's age range. In doing so, no one would be able to know and ascertain the original information contained in the dataset. Also, removing the gender, zip code and Id number of the users from the dataset would better enhance the privacy of the dataset.

TABLE X
KA-ANONYMIZED DATA

Notation	Meaning
Є	Differential Privacy
D	Database
C	Cut-off point
a_i	Output
I	Sensitivity
f_k	Halt
C	Count

Differential Privacy Algorithm
Input: private database D, an adaptively chosen stream of sensitivity L queries $f_1, ...$, a threshold T, and a cutoff point c.
Output: stream of answers $a_1, ...$

Sparse($D, \{f_i\}, T, c, \epsilon, \delta$)
If $\delta = 0$, **then**
let $\sigma = \frac{2c}{\epsilon}$.
else
$\sigma = \sqrt{32c\ln\frac{1}{\delta}}$
end if
Let $\hat{T}_0 = T + Lap(\sigma)$
Let count = 0
For each query i do
Let $v_i = Lap(2\sigma)$
If $f_i(D) + v_i \geq \hat{T}_{count}$ **then**
Output $a_i = T$
Let count = count + 1
Let $\hat{T}_{count} = T + Lap(\sigma)$
else
Output $a_i = \perp$
end if
if count $\geq c$ **then**
Halt.
end if
end for

Fig. 6 Differential Privacy algorithm [13]

We noted that DP has reduced data utility and requires a notice to be added to the area where the risk is higher. So, using KA and DP become very necessary as deduced from Table XI.

TABLE XI
DP ANONYMIZED DATA

Quasi-Identifier				Sensitive Identifier
Age	Gender	Zip code	Id	Disease
34**	*	*	*	COVID-19
69**	*	*	*	Ebola
76**	*	*	*	HIV
45**	*	*	*	Flu
90**	*	*	*	Diabetes
76**	*	*	*	Cholera

E. Proposed Hybrid Data Privacy Model

The combination of KA and DP results in a hybridized data privacy model as proposed in this study, gives a strong and effective data privacy model that will be used to protect the

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

confidentiality and integrity of sensitive IoT generated data. As discussed above and in previous chapters, this research proposes a hybrid data privacy model that combines both DP and KA data, privacy models. This is important to protect or hide individual information from every malicious person. The symbols used are defined in Table XII and the algorithm is captured in Fig. 7.

TABLE XII
DP ANONYMIZED DATA

Notation	Meaning
Ka	K-Anonymity
∈	Differential Privacy
OD	Original dataset
KQID	k-quasi
eQID	∈-quasi
Ec	Equivalence class

DPKA - Differential Privacy and k-Anonymity (k, ϵ)

Input: Dataset D

Output: (k, ϵ)-anonymous dataset D^*

1 Perform attribute classification;

2 kQID = k-quasis;

3 eQID = ϵ-quasis;

4 Remove explicit identifiers;

5 Perform k-anonymisation on kQID;

6 **for each** Equivalence class ec of the k-anonymous dataset **do**

7. $Apply\ \epsilon - differential\ privacy\ mechanism\ to\ eQID\ to\ the\ record\ of\ ec$

end for

8 Merge the ϵ-differentially private and k-anonymous equivalence classes;

9 Merge the ϵ-differentially private and k-anonymous equivalence classes;

10 Shuffle records;

Fig. 7 Hybrid Data Privacy algorithm [14]

The algorithm presented in Fig. 7 shows the combination of DP and KA data privacy models. Table XII presented the meaning of all the symbols and notations that were used in Fig. 7. The dataset has been input from the IoT devices, the first step that was carried out on the dataset was to do attribute classification after KQID and eQID were introduced. Also, all explicit identifiers were removed from the dataset before performing KQID, the conditional statement was meant to do for each equivalence class ec of the k-anonymous dataset to do eQID, then finally set to merge DP and KA equivalence classes and shuffle the dataset and repeat the process until the final output was produced as shown in Table XIII.

TABLE XIII
HYBRID DATA PRIVACY ANONYMIZED DATA

Quasi-Identifier				Sensitive Identifier
Age	Gender	Zipcode	Id	Disease
*	*	*	*	COVID-19
*	*	*	*	Ebola
*	*	*	*	HIV
*	*	*	*	Flu
*	*	*	*	Diabetes
*	*	*	*	Cholera

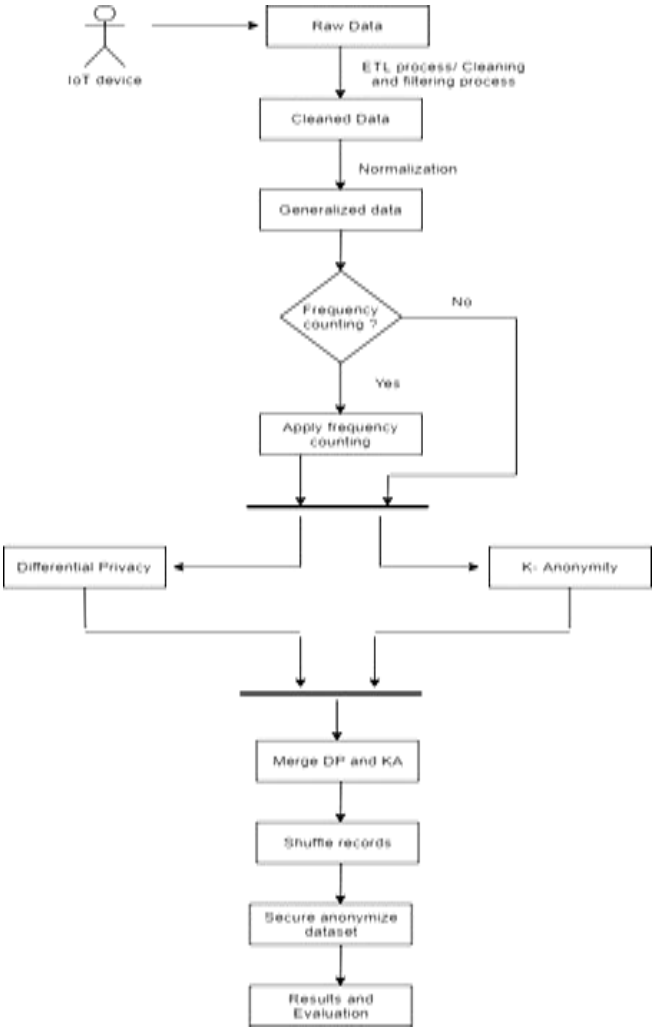

Fig. 8 Differential Privacy algorithm [13]

Fig. 8 shows how an IoT device generates data and the steps of how the data are being stored:

a. The IoT device sends the data out.
b. The researchers apply ETL (Extraction, Transformation and Loading) on the dataset when cleaning and filtering the dataset.
c. After the dataset is cleaned, we then normalized the dataset by applying the generalization and suppression method; that is, by editing the original dataset's actual attributes. We removed one or two attributes from the dataset which would now help in changing the original content of the dataset to further confuse the adversary from gaining access and meaning into what the dataset looks like and by

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

suppression, we removed some records from the datasets, which further helps to protect the dataset from unauthorized persons**.**

d. The frequently counted process was done for further analyses.
e. After going through the four steps stated above, we applied a double or merged data privacy model on the dataset (DP and KA).
f. The data were then shuffled to further confuse the adversary the more, before producing the final Real-Time Data Analysis Framework output.

This section discussed the proposed framework for classifying connected healthcare device systems using the best ML prediction model. The choice of the best algorithm in the framework was informed by the fact that it had the lowest train and test time. The framework would provide knowledge of the relationships and strengths among the patients with diabetes and blood sugar levels. Fig. 9 shows the proposed framework of the connected healthcare device system.

Components of the proposed framework are discussed as follows:

1) *Connected Devices:* The connected device described in vivid detail how the client's (the patient's) engagement with the medical staff's (doctors, nurses, and receptionists) as shown in Fig. 9.
2) *Decision Tree Model:* The decision tree enables us to decide what ML model is best suited for our trained data to make better predictions. It is a classifier expressed as a recursive partition of the instance space. It is a directed tree made up of a node that forms a rooted tree. It is made up of a root node and internal nodes. The model is designed to promptly detect those patients that are suffering from diabetes and high blood pressure when the various results are collected from all the medical sensor tools.
3) *Connected Devices:* All connected IoT devices are connected to the medical sensor through the Internet which then enables the data to be trained to make better predictions. It is a classifier expressed as a recursive partition of the instance space. It is a directed tree made up of a node that forms a rooted tree. It is made up of a root node and internal nodes. The model is designed to promptly detect those patients that are suffering from diabetes and high blood pressure when the various results are collected from all the medical sensor tools.
4) *Train Data:* The trained data are set to be the historical data used with test data to be able to decide how efficient our final output shall be with the decision tree model. The data are stored in the database and new features can be added at any time.

F. Benefits and Limitations

This section presented the benefits and limitations of connected healthcare system framework.

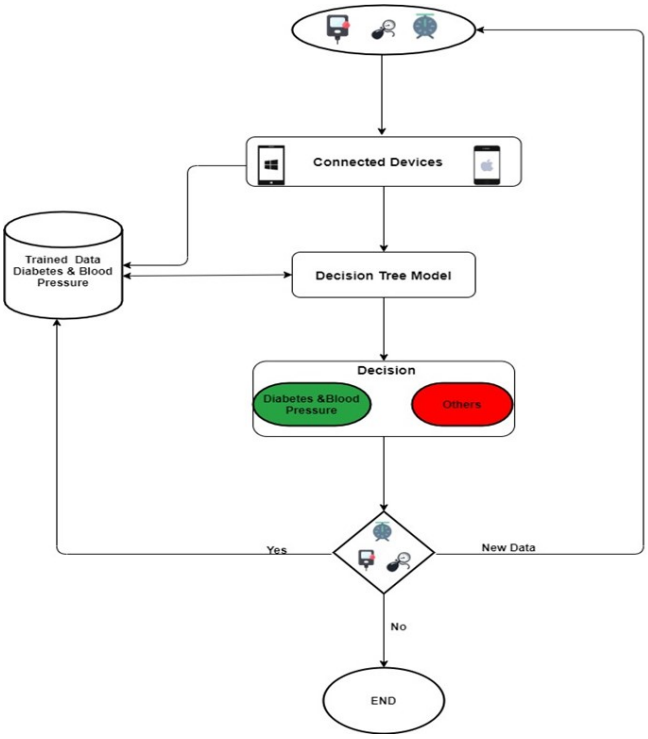

Fig. 9 Connected healthcare system framework

1) Benefits

With the explosion of digital technology, introducing a smart healthcare system, the system is set to capture and monitor all events on a real-time basis, and this better helps to facilitate a quicker and fast approach more promptly in attending to patients' needs. With the introduction of this system, costs would be reduced drastically, and better responses to urgent role calls by doctors and nurses can also be achieved.

2) Limitations of Connected Healthcare System Framework

One major concern that this kind of system has exposed us to is a breach of security and privacy of users because users' data are now there in the public domain, making it very easy for anyone to or be able to gain access if the dataset is not properly protected. Another limitation can also be the training of doctors and nurses on how to use the system.

III. Result and Discussion

We discussed the simulations carried out to evaluate the performance of cloud and fog computing and the obtained results. The system characteristics, evaluation parameters, and the procedure we used for the simulation were all described at the conclusion

The results were evaluated, and the performance of the fog storage system and cloud storage system were checked. A smart healthcare system was used as a case study to demonstrate how data can be sent into the fog system and cloud system through the devices such as smart headsets, etc., in real-time. Moreover, this research evaluated the time taken to complete each process as well as the network usage, the

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

latency and the amount of energy consumed. These are important factors that help determine which of the two platforms is cost-effective in the efficient management of the healthcare system's data. This was for the proper management of patients suffering from various chronic diseases like diabetes, whose historical dataset was used to demonstrate our result. In this study, the researchers conceptualized security to be data privacy, they adopted two data privacy models, and merged the two data privacy models to provide more protection to the dataset. The reason behind our choice was because one of the primary goals of this research, which was to suggest a more secure cloud storage platform. With the results obtained, this research suggests the healthcare system adopt a robust and cost-effective platform for their data storage. To perform the simulations, iFogSim software was used.

A. Simulations

This section presented the setup and discussed the parameters used in performing the simulations carried out in this research. An ultrasound IoT camera was used in the simulations. The system was installed in a healthcare clinic. The system was set up for use by medical personnel (doctors and nurses) when attending to sick patients to diagnose them. The system automatically takes the patients' details and sends the results into the storage platform. Accordingly, the ultrasound camera was set to send a real-time update to both the fog gateway and cloud data center simultaneously. It was set to send the updates in parallel into the storage system so that we could be able to determine the better method or approach to adopt. The simulation was repeated several times and evaluated based on the execution time i.e., the time taken for one complete simulation, network usage, which is the amount of network used in the process, and delay, or latency, which is the variance in sending and receiving periods in data transmission. That is, the results obtained were evaluated based on the metrics shown in Table XVI. The properties of the system used in performing the simulations are shown in Tables XIV-XVI.

TABLE XIV
SYSTEM PROPERTIES

Parameter	Value
Software	Linux-Ubuntu 18.08 LTS, 64-bit operating system
	iFogSim-master
	Eclipse IDE-2021-03-R-win32-x86_64
Hardware	Hard Disk 1Tetrabyte
	RAM 16G
	Processor: Intel® Core™ i7-45705 CPU @ 2.90G

TABLE XV
SIMULATION PARAMETERS

Parameter	Value
Simulator	iFogSim-master
Simulation Time	1541- 12748 seconds
Model	DPKA
Data transmission interval	0.1 seconds
File size	130 bytes
Number of Area (s)	1-4
Number of Headset Per Area(s)	4-12

TABLE XVI
EVALUATION METRICS

Metric	Description
Energy consumption	The total amount of energy consumes during the simulation period.
Execution time	The time it takes to complete the simulation.
Network usage	Network usage is generally the total amount of traffic on the network when linked with the highest amount that the network can support and each given process of the simulation.
Delay	The variance in sending and receiving periods in data transmission.

The various actors in the healthcare system are all users of the system. The ultrasound camera was the IoT device used when all various actors were attending to patients. The system was set to send information into the fog gateway and cloud data center. Then, IFogSim software was used in analyzing the data.

B. Results Analysis

This section presents the findings of the study obtained from the conducted simulations.

1) Execution Time

This subsection presents the execution time of each of the processes performed. Fig. 10 shows the result of execution time for both fog gateways and the cloud data center. The cloud data center took a long process to be completed compared to the fog gateway as shown in Fig. 10. It was obvious that far less time was needed to complete the process. The cloud systems show a high execution time range between 1617 sec to 10080 sec, and the fog system shows a low execution time range from 1353 sec to 9694 sec. The result implies that the cloud system requires more execution time than the fog system.

Fig 10. Execution time for both fog gateways and cloud system

2) Network Usage Execution Time

Fig. 11 shows the results of network usage between fog systems and cloud systems. From the results obtained, the fog system required low network usage that ranges between 10979 kilobytes to 183169 kilobytes than the cloud system which ranges between 166344 kilobytes and 1101319 kilobytes as shown in Fig. 11.

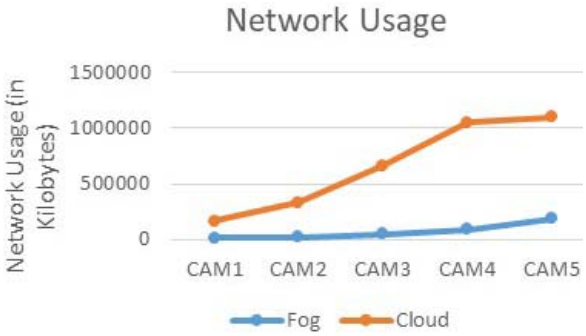

Fig. 11 Network usage between fog gateway and cloud system

3) Network Delay

Fig. 12 shows the network delay or the latency that occurs in the fog storage system and the cloud storage system. The cloud system shows high latency that ranges between 105253338801 and 311978448789, while that of the fog system shows low latency between 535714286 and 535714857. Based on the results, we deduced that the high latency problem is associated more with the cloud system while the fog system produced less delay when sending and receiving data. With these results, the cloud system cannot guarantee a fast response or is not responsive to a reactive system due to more delays in the process. Thus, this shows that the fog system is most suitable for a system that needs faster processing time.

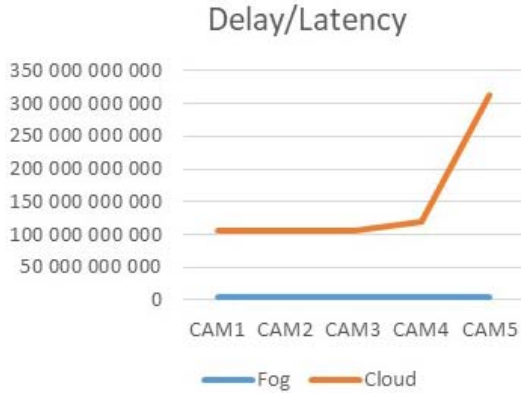

Fig. 12 Network delay between fog gateway and cloud

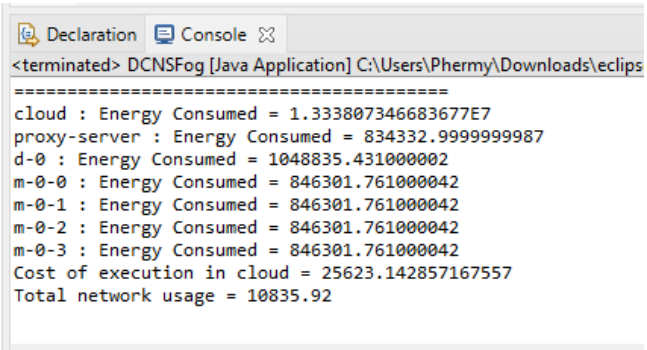

Fig. 13 Energy consumption

4) Energy Consumption

From the simulation results obtained, it was noted that more energy was consumed in the cloud data center system than that of the fog system. For the cloud data center, the total energy consumed was 1.333807346683677E7 when compared to that of the fog system which is 834332.9999999987. This is shown in Fig. 13, and this suggests the use of a fog system over a cloud data center. The cloud data center consumed more energy, also the cost of processing data was more in the cloud system than on the fog system.

C. Discussion and Comparison

This paper has presented the results of the simulations performed to access the cost-effectiveness of fog and cloud data centers in storing IoT data. The results show that the fog system used less execution time when compared with that of the cloud system. It also shows that the cloud data center system used more bandwidths than the fog system. Moreover, the fog system also produced less latency compared to the cloud data center system as well as energy consumption. Figs. 11-13 present the results of the fog system, and show that the fog system is more suitable for storing IoT generated data. This implies that, with the fog system, the process is brought closer to the device than that of the cloud data center which makes the fog system preferable to the cloud data center system.

Based on the results, this study undoubtedly shows us that fog systems perform far better than cloud systems considering execution time, network usage, and network delay and energy consumption. Table XVII gives a comparison of this research's findings with related research in the literature. It was observed that the simulation results corroborate with findings of other people [4]-[10]. Some similarity was also observed in the results from previous studies which makes this research very consistent with previous studies [15], [16]. This study attempted to improve on some of the areas that could better help in providing a more protected platform to secure users' sensitive data when stored in either the fog system or cloud system.

IV. Conclusion

In summary, from the results as presented in this paper, we were able to evaluate the effectiveness of storage architecture to show the advantages and disadvantages of using fog storage systems and cloud storage systems. Hence, our results can convincingly show that the use of fog storage systems is better and managing all IoT devices data. Also, merging of two data privacy models namely, K-Anonymity and Differential Privacy was able to procure effective data privacy. It was also deduced that the merging of two or more data privacy models as illustrated in this paper will further help in providing efficient protection to IoT generated data when stored. It was also recommended that a data privacy model that can proffer secured protection to both cloud and fog architecture can be developed.

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

Open Science Index, Industrial and Manufacturing Engineering Vol:17, No:7, 2023 publications.waset.org/10013174.pdf

TABLE XVII
EVALUATION METRICS

Ref.	Attribute	Cloud Datacenter	Fog Gateway
[17]	Latency	-	-
	Network Usage	High	Low
	Execution Time	High	Low
	Energy Consumption	-	-
	Distribution	Centralized (One server)	Distributed (Many nodes)
	Deployment	More costly to deploy	Less costly to deploy
	Security	-	-
[18]	Latency	High	Low
	Network Usage		
	Execution Time		
	Energy Consumption	High	Low
	Distribution	Centralized (One server)	Distributed (Many nodes)
	Deployment	More costly to deploy	Less costly to deploy
	Security	-	-
[19]	Latency	High	Low
	Network Usage	High	Low
	Execution Time	-	-
	Energy Consumption	-	-
	Distribution	-	-
	Deployment	-	-
	Security	-	-
[20]	Latency	High	Low
	Network Usage	High	Low
	Execution Time	High	Low
	Energy Consumption	-	-
	Distribution	-	-
	Deployment	-	-
	Security	Low	High
[21]	Latency	-	-
	Network Usage	High	Low
	Execution Time	High	Low
	Energy Consumption	High	Low
	Distribution	-	-
	Deployment	-	-
	Security	-	-
[22]	Latency	High	Low
	Network Usage	-	-
	Execution Time	-	-
	Energy Consumption	High	Low
	Distribution	Centralized (One server)	Distributed (Many nodes)
	Deployment	More costly to deploy	Less costly to deploy
	Security	Low	High
[23]	Latency	High	Low
	Network Usage	-	-
	Execution Time	-	-
	Energy Consumption	High	Low
	Distribution	Centralized (One server)	Distributed (Many nodes)
	Deployment	-	-
	Security	-	-
Our Model	Latency	High	Low
	Network Usage	High	Low
	Execution Time	High	Low
	Energy Consumption	High	Low
	Distribution	Centralized (One server)	Distributed (Many nodes)
	Deployment	More costly to deploy	Less costly to deploy
	Security	-	-

ACKNOWLEDGMENT

The authors acknowledged the resources and financial support made available by Department of Information Technology Systems, Walter Sisulu University, Eastern Cape, South Africa.

REFERENCES

[1] Botta, A., et al., *Integration of cloud computing and internet of things: a survey*. Future generation computer systems, 2016. 56: p. 684-700.
[2] Jiang, C., et al., *An edge computing platform for intelligent operational monitoring in internet data centers.* IEEE Access, 2019. 7: p. 133375-133387.
[3] Okay, F.Y. and S. Ozdemir. *A fog computing based smart grid model*. in *2016 international symposium on networks, computers and communications (ISNCC)*. 2016. IEEE.
[4] Dastjerdi, A.V., et al., *Fog computing: Principles, architectures, and applications*, in *Internet of things*. 2016, Elsevier. p. 61-75.
[5] Gupta, H., et al., *iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments.* Software: Practice and Experience, 2017. 47(9): p. 1275-1296.
[6] Abdallah, M., et al., *Delay-sensitive video computing in the cloud: A survey.* ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2018. 14(3s): p. 1-29.
[7] Popli, S., R.K. Jha, and S. Jain, *A survey on energy efficient narrowband internet of things (NBIoT): Architecture, application and challenges.* IEEE Access, 2018. 7: p. 16739-16776.
[8] Alexandru, A., D. Coardos, and E. Tudora. *IoT-Based Healthcare Remote Monitoring Platform for Elderly with Fog and Cloud Computing*. in *2019 22nd International Conference on Control Systems and Computer Science (CSCS)*. 2019. IEEE.
[9] NICOLAU, D.N., A. Alexandru, and M. Ianculescu, *An IoT, Virtual Machines and Cloud Computing-based Framework for an Optimal Management of Healthcare Data Collected from a Smart Environment. A Case Study: RO-Smart Ageing Project.* Informatica Economica, 2019. 23(3).
[10] Jain, P., M. Gyanchandani, and N. Khare, *Big data privacy: a technological perspective and review*. Journal of Big Data, 2016. 3(1): p. 1-25.
[11] Yu, S., *Big privacy: Challenges and opportunities of privacy study in the age of big data.* IEEE access, 2016. 4: p. 2751-2763.
[12] Salas, J. and V. Torra, *A general algorithm for k-anonymity on dynamic databases*, in *Data privacy management, cryptocurrencies and blockchain technology*. 2018, Springer. p. 407-414.
[13] Dwork, C. and A. Roth, *The algorithmic foundations of differential privacy.* Foundations and Trends in Theoretical Computer Science, 2014. 9(3-4): p. 211-407.
[14] Holohan, N., et al., *($ k $, $\epsilon $)-Anonymity: $ k $-Anonymity with $\epsilon $-Differential Privacy*. arXiv preprint arXiv:1710.01615, 2017.
[15] Mahmud, R. and R. Buyya, *Modelling and simulation of fog and edge computing environments using iFogSim toolkit.* Fog and edge computing: Principles and paradigms, 2019: p. 1-35.
[16] Bala, M.I. and M.A. Chishti. *Offloading in cloud and fog hybrid infrastructure using iFogSim*. in *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. 2020. IEEE.
[17] Dhingra, S., et al., *Internet of things-based fog and cloud computing technology for smart traffic monitoring.* Internet of Things, 2020: p. 100175.
[18] Baucas, M.J. and P. Spachos, *Using cloud and fog computing for large scale iot-based urban sound classification.* Simulation Modelling Practice and Theory, 2020. 101: p. 102013.
[19] Sunyaev, A., *Fog and edge computing*, in *Internet Computing*. 2020, Springer. p. 237-264.
[20] Wang, T., et al., *Data collection from WSNs to the cloud based on mobile Fog elements.* Future Generation Computer Systems, 2020. 105: p. 864-872.
[21] Chaurasia, N., et al., *Comprehensive survey on energy-aware server consolidation techniques in cloud computing.* The Journal of Supercomputing, 2021: p. 1-56.
[22] Linthicum, D.S., *Connecting fog and cloud computing.* IEEE Cloud Computing, 2017. 4(2): p. 18-20.
[23] Bonomi, F., et al. *Fog computing and its role in the internet of things*. in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. 2012.