Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.031283
Article

Tech Science Press

Health Data Deduplication Using Window Chunking-Signature Encryption in Cloud

G. Neelamegam* **and P. Marikkannu**

Department of Computer Science and Engineering, Anna University Regional Campus Coimbatore, Coimbatore, 641046, India
*Corresponding Author: G. Neelamegam. Email: gneelamegam21@outlook.com
Received: 14 April 2022; Accepted: 25 May 2022

Abstract: Due to the development of technology in medicine, millions of health-related data such as scanning the images are generated. It is a great challenge to store the data and handle a massive volume of data. Healthcare data is stored in the cloud-fog storage environments. This cloud-Fog based health model allows the users to get health-related data from different sources, and duplicated information is also available in the background. Therefore, it requires an additional storage area, increase in data acquisition time, and insecure data replication in the environment. This paper is proposed to eliminate the de-duplication data using a window size chunking algorithm with a biased sampling-based bloom filter and provide the health data security using the Advanced Signature-Based Encryption (ASE) algorithm in the Fog-Cloud Environment (WCA-BF + ASE). This WCA-BF + ASE eliminates the duplicate copy of the data and minimizes its storage space and maintenance cost. The data is also stored in an efficient and in a highly secured manner. The security level in the cloud storage environment Windows Chunking Algorithm (WSCA) has got 86.5%, two thresholds two divisors (TTTD) 80%, Ordinal in Python (ORD) 84.4%, Boom Filter (BF) 82%, and the proposed work has got better security storage of 97%. And also, after applying the de-duplication process, the proposed method WCA-BF + ASE has required only less storage space for various file sizes of 10 KB for 200, 400 MB has taken only 22 KB, and 600 MB has required 35 KB, 800 MB has consumed only 38 KB, 1000 MB has taken 40 KB of storage spaces.

Keywords: Health data; encryption; chunks; cloud; fog; deduplication; bloom filter

1 Introduction

Cloud-Fog comprises of various sensor devices for sharing health care data that are remotely communicated through different networks. In addition to that, these sensor devices are connected via Fog server-based cloud-sever to transmit the data. The cloud server is the primary source of storing the data for preserving the privacy and data security and also, it is effective in cost [1]. Therefore, the protection of health-care data and maintenance of confidentiality on the patient information are essential [2]. In the health care data sector, storing the duplicated patient data will produce the wrong prediction of disease.

Also, sharing or uploading the same health data happens on demand in data storage. This duplication of health-related data needs to be detected and removed from the database because it occupies unnecessary storage space.

In order to protect the privacy of medical data, accessing the control in an attribute-based forward mechanism is to be implemented. Moreover, it is more flexible in accessing the control mechanism, scalability, accessing fine granularity in the cloud storage environment. On the contrary, some of the issues in this attribute-based access control are low efficiency in data encryption, retrieval strategy, lack of continuity, conflict in the detection, etc. [3]. Handling a massive volume of health care data is a challenging task. To manage the data, deduplication is the best technique because it eliminates the duplicate copy of data and only maintains a single copy of it. Similarly, it minimizes the utilization of storage space and maintenance cost of replicated data. The detection of redundancy data using Content-Defined Chunking (CDC) plays an important role [4]. Deduplication of data can either work at block level or file level. At the file level, it verifies that the content of the two files is the same. If so, it eliminates one file and keep another file in a secured mode. In the same way, the block level searches the content of the block, eliminates one copy of the league, and retains another block. The block-level of the file involves four processing steps; chunking, fingerprinting, indexing of fingerprints, and managing the stored information of data [5,6].

The primary goal of this proposed work is to secure the health care data deduplication model for cloud-fog storage integrated environment. Many research works have been done in the deduplication of health care data. However, the issues in the existing traditional algorithms are low efficiency in detecting the redundancy of data and storage space. The proposed work implements the deduplication data using a window size chunking algorithm with a biased sampling-based bloom filter. It provides a secured health data using the Advanced Signature-Based Encryption (ASE) algorithm in the Fog-Cloud environment (WCA-BF + ASE). The major contributions of the proposed work are:

1. Implementation of window size chunking algorithm for splitting the stream of health care data.
2. To eliminate the redundancy of data using Bloom filter.
3. To provide high security and preserve privacy information in the cloud storage using an advanced signature-based encryption algorithm.

The paper has been organized as follows; Section 2 describes the literature review, Section 3 describes the deduplication using WCA-BF + ASE, Section 4 discusses the experimented results, and Section 5 concludes the paper with future directions.

2 Review of Literature

A massive volume of health domain data is generated daily and stored in the cloud server. In order to maintain the medical records in an effective way and to increase the quality of patient care, de-duplication is implemented. Health care data contains vital information about the patient, and it is used to detect the disease at an early stage and diagnose the condition accurately. Sophisticated software has been developed for further data processing to improve the quality effectively. It is used to maintain the digital records of heath details inaccurately. Remotely, the health care data management allows the authenticated users to securely access the healthcare data in the cloud environment.

Similarly, the healthcare practitioners can access the cloud storage environment to store the health information of the patients [7]. The proposed de-duplication is based on the concept of ABE, which contains the health care information and protects the storage space. It uses various ABE techniques for the elimination of de-duplication [8]. Yang et al. [9] presented a customer Side Scrambled De-duplication plan and developed a plot in a safe and proficient method that produces machine learning-based

encryption keys. The advantage of this technique is referencing the customer-side rate constraints and diminishing both the capacity and the de-duplication of data. Xuan et al. [10] mentioned about a secured approach for accessing the control of a massive volume of data stored in a well-secured structure and implementing the de-duplication process to remove the duplicated health care data. The survey of various existing techniques is stated in Tab. 1.

Table 1: The survey on the de-duplication of data processing in a secured cloud storage

Author	Contribution	Analysis
Chen et al. [11]	Cost-effective convolutional neural nets training based on image deduplication	Street View House Numbers (SVHN)
Lu et al. [12] (2021)	Image deduplication based on hashing and clustering in cloud storage	Hashing, clustering, local binary pattern
Kharat et al. [13]	It is used in IoT-enabled heart disease prediction a system using the modified deep convolutional network classifier. It maintains security and privacy-preserving of health-related information	Secure, privacy-preserving
Xia et al. [14]	Detection of duplication attack in the blind forgery using SIFT technique	Simulation time
Maqdah et al. [15]	Content-defined chunking for data deduplication based storage systems	Chunking
Nie et al. [16] (2020)	Deduplication is based on content-aware	Storage space
Yigzaw et al. [17] (2019)	De-duplication technology based on optimized blocking algorithm	Secure, storage space

3 Proposed Methodology

This proposed work reduces the decloud-fog-based health care data and securely stores it. It is composed of two phases;

Phase 1: Eliminates the duplicated health care data and maintain the data as a single copy by using a window size chunking algorithm

Phase 2: Storing the data in the cloud-fog-based environment securely by using the Advanced Signature-Based Encryption (ASE) algorithm. The workflow of this proposed work is given in Fig. 1.

Fig. 1 illustrates the data that is collected from various sensor devices of the patient, and it is transmitted to data users. Then the duplicated information is eliminated by using the deduplication procedure of using a window size chunking algorithm with a biased sampling-based bloom filter. After the elimination process, it moves to the FoG server. The FoG server processes the data based on the cloud format and uploads the secure data by applying the Advanced Signature-Based Encryption (ASE) algorithm.

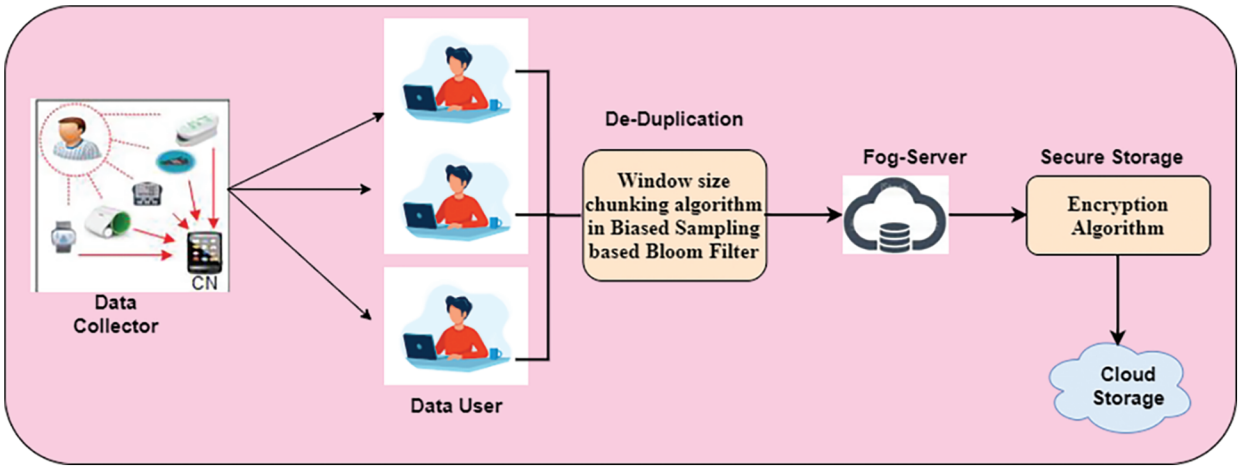

Figure 1: Workflow of the proposed work

3.1 De-Duplication

A window size chunking algorithm with a biased sampling-based bloom filter is implemented in the proposed work to eliminate the de-duplicated data. This de-duplication process is implemented using two stages;

Stage 1: Splitting of health data based on window size using Window Size Chunking Algorithm (WCA).

Stage 2: Based on the input, the window size chunks value with biased sampling-based Bloom Filter (BF) is calculated. Fig. 2 shows the De-Duplication of health care data.

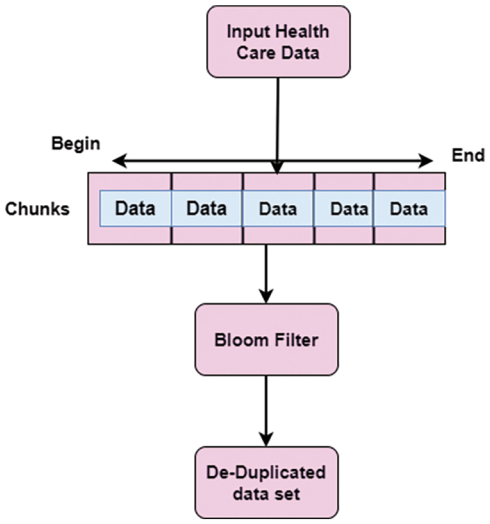

Figure 2: De-duplication process

3.1.1 Window Size Chunking Algorithm (WCA)

It controls the chunk size based on the threshold value for the size of the window. The primary function of the FoG server is to manage the data chunks, deduplicate the health data, and store it in the cloud. Data is read from various sensor devices of the patient in the string format and separated by ":". For every patient, the

complete information is separated by ";". For patient A, data is collected in terms of temperature, blood sugar, heart rate, and cholesterol in the form of 100.4:240:90:123; here, the record is ended by a semicolon. It refers to the single entity of a patient. Algorithm 1 shows the Window Size Chunking Algorithm (WCA).

Algorithm 1: Window-Size Based Chunking Algorithm

Input: Health care data in a string format *str_hcd*

Output: Chunk of health care string data in a window size *wind*

Predefined Value: Delimiter (;)

Step 1: $Spilt\_str = split(str\_hcd,\ D)$

Step 2: for $i = 0$ to $Split\_str.size$

Step 3: if $(selected\_str.size > wind)$

Step 4: $selected\_str.remove(selected\_str.size - 1)$

Step 5: Return *selected_str*

Step 6: Break

Step 7: Else If $(selected\_str.size == wind)$

Step 8: Return *selected_str*

Step 9: Break

Step 10: Else If $(selected\_str.size > \frac{3}{4} wind)$ && $(selected\_str.size < wind)$

Step 11: Return *selected_str*

Step 12: Break

Step 13: Else

Step 14: $selected\_str[i] = split\_str[i]$

Step 15: End If

Step 16: End For

In Algorithm 1, the health care data is taken as input and in string format, each patient record is separated by delimiter (;) and stored it in array format of string type as $split\_str[i]$ for each index value. Using FOR loop, it traverses all medical data items. If the health care data of the patient doesn't exceed the window size of greater than 75% of medical data, it shows that it is stored in another array. Selecting the chunk size of the window is based on three conditions namely, if array size is greater than window size $(wind)$, it deletes the last item and returns the array. Secondly, if array size is equal to window size $(wind)$, it returns the array value as it is. Finally, an array size is selected based on the selected data in the range between ¾ and its window size. It returns the selected range of chunks.

3.1.2 Biased Sampling-Based Bloom Filter

This technique is used for de-duplication of huge volume of stream health care data from the output of Algorithm 1. Bloom filters B is an array size of n with b bits. Initially it is assigned as 0. The insertion of new element in the bloom filter and its membership of data element is $m \in M$. In each bloom filter, a new data element de from the array and it is hashed into b bits with the help of m in different uniform hash functions which are randomly selected. The randomly selected p independent hash functions are defined as $hash_h(.)$ and $1 \leq h \leq p$. Implementation process of bloom filter in the detection of de-duplication is given below.

- At first, the hash function $hash_h(de)$ to the data element d is evaluated.
- To get position pos in the array n using modulo function md, it is applied to all elements. Then the bloom filter is defined as $bf_h(de) = hash_h(de) mod md$, where $bf_h(x) \in [0,\ md-1]$.
- To add new data element d and verified by its position of b bits whether it is set or not. Once the data element is inserted, all b bit position elements are set to 1, then it is considered as insertion process.
- The data element de is eliminated from bloom filter and its position is set to 0.

The bit position of the existing element is checked, whether it has value 1, then the bloom filter considers that element as duplicate or else, it is distinct.

Here is the current stream length and the size of the element in the bloom filter. In the bloom filter, the number of bits increases to set, and it also increases the false positive rate and for the distinct element, it is falsely reported as duplicate. If the length of the healthcare data stream increases and the probability of the same element rate also increases. Inserting aspects from the health care data stream in the bloom filter sets all bits into 1. That is a false positive rate. To overcome this drawback, if a new element is inserted into the array of bloom filters, it randomly deletes a component and sets the value to 0. It creates false negatives, and its duplicate value is treated as distinct. Algorithm 2 implements the concept of de-duplication detection.

Algorithm 2: De-Duplication using Biased Sampling-Based Bloom Filter

Input: Chunk of health care string data

Output: Detection of Duplicate health care data

Step 1: Evaluate the value of bloom filters B in FPR_{de}

Step 2: Build bloom filters B with bits of memory

Step 3: $itera \leftarrow 1$

Step 4: For each data element d of n do

Step 5: $hash_h(de)$ into b bit position, $hash_h(de) = hash_1,\ \ hash_2,\ \ hash_3,\ \ldots.,\ hash_b$

Step 6: If all bit position in $hash_h(dee) = 1$ tehn

Step 7: $out \leftarrow duplicate$

Step 8: Else

Step 9: $out \leftarrow distinct$

Step 10: Endif

Step 11: if $itera \leq de$

Step 12: set all data element's position of bit in de

Step 13: Else

Step 14: If $\left(\frac{de}{itera}\right) \leq thresh$ then

Step 15: for all bit position in $hash_h$ do

Step 16: If $hash_{de} = 0$ then

Step 17: Identify a bit in j^{th} bloom filter is set to 1 and reset to 0.

Step 18: Set the bit at hp_i position to 1

Step 19: End IF

Step 20: End For

If the distinct data element of the health care data is reported as duplicate, it is False Positive Rate (FPR). But if False Negative Rate (FNR) occurs, duplicate health stream data is reported as distinct. In the Algorithm 2, de-duplication of health care data elements is reported in chunks. Let us consider the data elements of de_{m+1} and $(m+1)^{th}$, data element of chunk stream and applying hash function using Step 5. The bloom filter report is duplicate in case it satisfies as in Step 6 else, it is considered as distinct.

3.2 Encrypted Data Storage

In order to save the health care data in a secured way, this proposed Advanced Signature-Based Encryption (ASE) scheme is implemented. ASE scheme protects the privacy of the user's data. It allows only the authorised person to view the user's data. By using ASE signature scheme, the user's data can be accessed in the cloud and sent back to the individual patient. This health care data storage model is shown in Fig. 3.

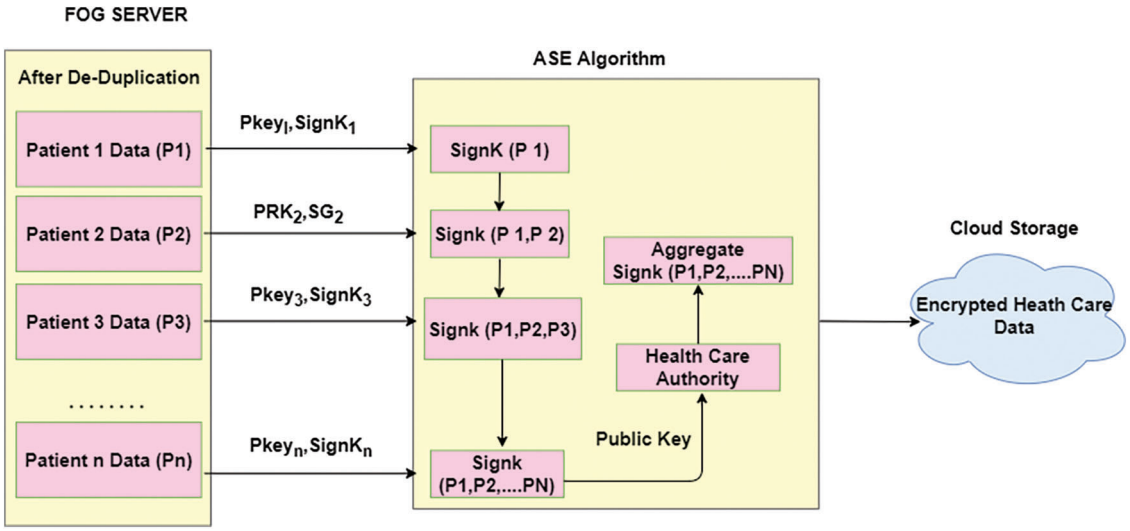

Figure 3: Health care data secure storage

E-Health care [17] data management has collected all the data from different sources. Then each data is divided into $(hc_1,\ hc_2,\ hc_3,\ \ldots,\ hc_n)$ along with patients' private key pair of $(prik_i,\ sig_i)$. To store the data from fog server to cloud storage, $patient_1$ using their private key $prik_1$ and sign key sig_1 and gets the signature sign $(patient_1)$ using ASE scheme. Similarly, $(patient\ _2)$ using their private key $prik_2$ and sign key sig_2 will get the signature sign $(patient_1,\ patient\ _2)$ using ASE scheme and so on. In order to get the aggregate sign, the public key $pubk$ of Patient Authority (PA) is added. Only then, the health care data will be sent to the cloud storage. This is implemented by Algorithm 3.

Algorithm 3: Generation of Key

Step 1: Using Algorithm 2, eliminates the duplicate data

Step 2: Generate the signature key value sig_i.

Step 3: Aggregate the health care data with the input health data, private key $prik_i$, sig_i.

Step 4: $dv \leftarrow Patient_i(healthdata_i,\ prik_i,\ sig_i + pubk_i)$

Step 5: If patient P_i is confirmed, the storage gets over to cloud storage.

It performs the encryption of health care data using hash function. The key pairs of public and private key of asymmetric scheme are used in this Algorithm 4.

Algorithm 4: Advanced Signature-Based Encryption Algorithm (ASE)

Step 1: Function Encryption $(dd\_healthdata_i)$

Step 2: Create Symmetry key $symmk_i$

Step 3: $encryppt \leftarrow (medical\ data_i,\ \ symmk_i)$

Step 4: *cipher text* $(CT) \leftarrow encrypt_{symmk}(dd\_healthdata,\ \ symmk_i)$

Step 5: *cipher key* $(CTK_i) \leftarrow (symmk_i,\ \ pubk_i)$

Step 6: End Function

Step 7: function $signature\ (dd\_health\ data_i)$

Step 8: Create asymmetric key pair $(askpub_i,\ \ askpri_i)$

Step 9: $hashkey_i \leftarrow estimate\ hash(dd\_health\ data_i)$

Step 10: using $prik_i,\ sig_i, hashke_i$ design a signature for the patient $Patient_i$

Step 11: Use Public key (pbk_i)

Step 12: End Function

The Algorithm 5 performs the asymmetric decryption of the health data in the cloud server using private key and a symmetric key.

Algorithm 5: Advanced Signature-Based Decryption Algorithm (ASD)

//CT-cipher text, CTEK- cipher Text key, prik-private key, symmk-symmetric key

Step 1: Function $decryption(CT,\ CTEK,\ prik,\ symmk)$

Step 2: $symmk \leftarrow decryption_{asymmk}(CTEK,\ prik)$

Step 3: $dd\_healthdata \leftarrow decryption(CT,\ symmk)$

Step 4: End function

For the generation of signature using ASE with verifying signature, in Algorithm 5, the digital signature by using sequential aggregate signature scheme is generated. It gives more security and less storage space. the concept of advanced signature-based algorithm is implemented. This algorithm further classified into two concepts, namely encryption and decryption which is implemented in Algorithms 4 and 5.

3.3 Verification of Digital Signature

The Algorithm 6 performs the identification of various sensor devices by "ID". In order to issue the certificate of verification and authentication, private keys of sensor devices with generation of random numbers are used. The mapping function is conducted by the cloud server for identification of sensor

device for the particular patient. The mapping of device ID for healthcare is mentioned. This is implemented by Algorithm 6.

Algorithm 6: Verification of Digital Signature

Step 1: For each sensor device $scert_i$ is used

Step 2: Cloud server-basedfog system is created.

Step 3: if $(healthdata = \ sensitivedata) then verify the data$

Step 4: elseif $(healthdata == non\_sensitivedata)$

Step 5: send health data to cloud from fog server.

Step 6: For each patient data $(Patient_i) dosensordevice \leftarrow scert_i$

Step 7: $scert_i + timst \leftarrow Patient_i$ //certificate and time stamp are assigned for each patient.

Step 8: function $authentication(healthdata)$

Step 9: Generate $prik_i$

// Certificate in sensor device $(scert_i)$ is match with certificate in cloud server.

Step 10: if $scert_i == scert_i incloudserver$ then

Step 11: Mutual authentication $(sedevice_1, sedevice_2 \ , \ Patient_i, \ Doctor_i)$ // P_i- Patient, D_i- Doctors

Step 12: **Function** $verification(CT, signpubk)$

Step 13: Calculate the hash function for medical data of patient from sensor device using signed public key $(signpubk)$ of sensor device.

Step 14: compare hash value of patient form sensor device with hash value in cloud server.

Step 15: if it matches, "signature is correct", Else "signature is not correct"

Step 16. End if

Step 17: end function

Algorithm 6 describes the module that based on the verification of the signature, it will permit the user to access the health care data.

4 Result and Discussion

The proposed work WCA-BF + ASE is implemented from scratch which includes the process of chunking health related stream of data, removal of duplicated data, storing data in a secured way. The implantation of deduplication technology is deployed in Python 3.7, Tensorflow/Keras. The database is used in Amazon S3 (Cloud Storage). For the implementation of ASE, in the aspects of security concept based on Encryption time (Et), Decryption time (Dt), Time for key generation and security analysis are shown below. The unstructured health care dataset is encrypted and AWS creates a memory in the cloud storage. The sample dataset contains 3245 records with 30 features. This record also contains the duplicated records. After implementing the de-duplication, it contains 3100 records with 30 features [18]. The efficiency of implementing this WCA-BF + ASE algorithm with different size of data is discussed below.

4.1 Execution Time

Tab. 2 describes the execution time of various file sizes of data before applying de-duplication of health care data.

Table 2: Execution time for before applying de-duplication health care data

Execution time in (ms)	De-duplication file size (MB)				
Technique name	200	400	600	800	1000
Window Size Chunking Algorithm (WSCA) [1]	1200	3600	5000	7500	9300
Two Thresholds, Two Divisors (TTTD) [1]	1350	3780	5400	7800	9800
Optimal Removal of Deduplication (ORD) [2]	1175	3450	4800	6400	8900
Bloom Filter (BF) [18]	1190	3400	4725	6325	8650
WCA-BF + ASE (proposed)	975	3100	4600	5900	8400

In the observation of Tab. 2, the operation time of before applying de-duplication using various techniques like WSCA, TTTD, ORD, BF and WCA-BF + ASE in different size of files is illustrated. The proposed work requires less operation time for various files of 200 MB needs 975 ms, 400 MB needs 3100 ms, 600 MB file requires 4600 ms, 800 MB file requires 5900 ms and for 1000 MB needs 8400 ms. Tab. 3 describes the Execution time of various file size of data after applying de-duplication of health care data.

Table 3: Execution time for after applying de-duplication health care data

Execution time in (ms)	After de-duplication file size (MB)				
Technique name	200	400	600	800	1000
Window Size Chunking Algorithm (WSCA)	1100	3550	4970	7370	9280
Two Thresholds, Two Divisors (TTTD)	1320	3720	5370	7750	9750
Optimal Removal of Deduplication (ORD)	1150	3370	4650	6350	8760
Bloom Filter (BF)	1160	3300	4625	6225	8450
WCA-BF + ASE	875	2900	4450	5700	8320

Tab. 3 illustrates the operation time of before applying de-duplication using various techniques like WSCA, TTTD, ORD, BF and WCA-BF + ASE in different size of files. The proposed work requires less operation time for various files of 200 MB needs 875 ms, 400 MB needs 2900 ms, 600 MB file requires 4450 ms, 800 MB file requires 5700 ms and for 10000 MB needs 8320 ms.

4.2 Operation Time

Tab. 4 describes the operation time of various operations in the implementation of the proposed work.

Tab. 4 shows various operation time (ms) of window chunking stream of health data using Algorithm 1, Generation of key using Algorithm 3, Encryption using Algorithm 4 and decryption using Algorithm 5 and finally for the operation of verifying the signature by using Algorithm 6. Here before deduplication of data and after deduplication of data is analysed. After deduplication of data, it gives better performance. The

Encryption time is calculated by the difference between ending time of encryption with its beginning time and it is defined as:

Table 4: Operation time of the proposed work

Operation time (ms)	WCA-BF + ASE (Proposed)	
Technique name	Before de-duplication	After de-duplication
Window chunking stream of health data	5	4.2
Generation of key	4.5	3.75
Encryption	5.1	4.3
Decryption	0.8	0.5
Verification of digital signature	3.1	2.8

$$Encrypt_T = encrypt_{end} - encrypt_{start} \tag{1}$$

Here, $Encrypt_T$ is the encryption time, $encrypt_{end}$ is the ending time of encryption, $encrypt_{start}$ is the starting time of encryption. Fig. 4 shows the encryption time of various file sizes using various techniques that are applied after de-duplication is involved.

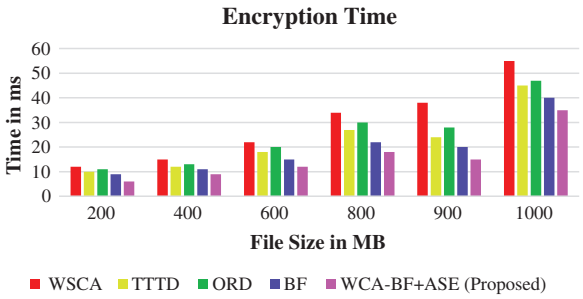

Figure 4: Encryption time

Fig. 4 presents the encryption time of various file sizes varying from 200 to 1000 MB.

Here, 200 MB in each iteration is increased. The proposed work WCA-BF + ASE produces better performance compared to the other existing algorithms like WSCA, TTD, ORD, BF.

$$Decrypt_T = Decrypt_{end} - Decrypt_{start} \tag{2}$$

Fig. 5 shows the decryption time of various file sizes using various techniques that are applied after de-duplication is involved.

Fig. 5 shows that the proposed work WCA-BF + ASE has produced better performance compared to the other existing algorithms like WSCA, TTD, ORD, BF.

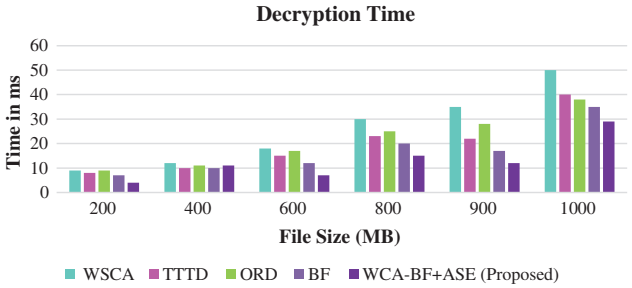

Figure 5: Decryption time

4.3 Security in Cloud Storage

Security is highly needed thing [19–20] in the cloud storage for the health care data. The level of security is evaluated by the hacked health care data with its original stream of health care data. It is defined as:

$$security\ level = \frac{Hacked_{data}}{Original\ _{data}} \tag{3}$$

Fig. 6 shows the comparison of various security level with the existing algorithm.

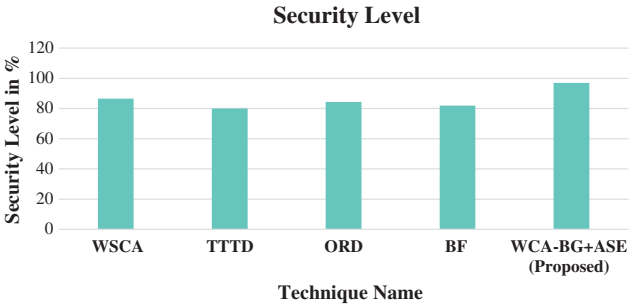

Figure 6: Security level

Fig. 6 describes the security level of various technique in the cloud storage of heath care data that is stored. Comparing it with the other existing techniques (WSCA got 86.5%, TTTD 80%, ORD 84.4%, BF 82%), the proposed work has got better security storage of 97%.

4.4 Energy Efficiency

The health care data management requires an authentication for accessing the data from the cloud storage as shown in Fig. 7. The average energy efficiency parameter of (bits/sec/joule) verifies each file in the cloud storage with the threshold value of 4 joules.

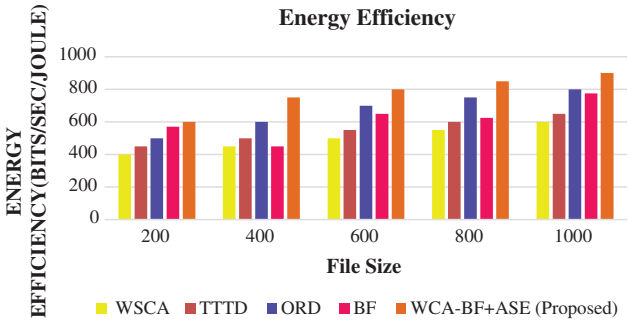

Figure 7: Energy efficiency

4.5 Throughput

In this performance parameter, it is the rate at which the de-duplication of health care data are transmitted to the cloud storage.

$$transaction\ per\ file = \frac{file\ size}{average\ transaction\ size} \tag{4}$$

$$fraction\ of\ file\ per\ second = \frac{1}{Node\ time\ in\ seconds} \tag{5}$$

$$transaction\ per\ file = transcation\ per\ file * fraction\ of\ file\ per\ second \tag{6}$$

This throughput parameter is compared with various techniques like WSCA, TTTD, ORD, BF and WCA-BF + ASE in different size of files. Fig. 8 shows the comparison of throughput.

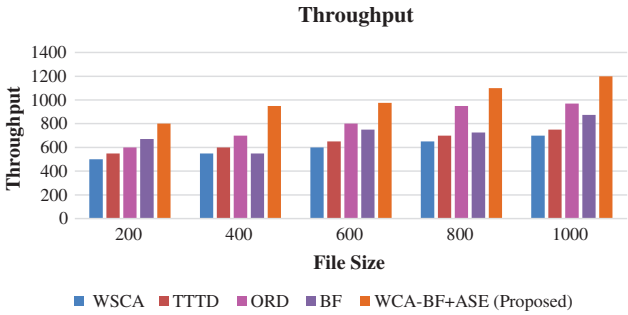

Figure 8: Throughput

In the observation of Figs. 7 and 8, the proposed method WCA-BF + ASE outperforms very well. Its average energy efficiency has increased the health care data transmission among the files in the cloud storage.

4.6 Storage Space in Cloud Environment

To protect the health care data, privacy preserving scheme is implemented to secure the data. Improving the security constraints for every file in the cloud storage, encryption and decryption techniques are applied in the data set after applying the de-duplication process.

The encryption cost of files is reduced and it increases the storage space. For all the encrypted files a secret key is generated. Without this secret key, the file cannot be accessed and also decrypted. In uploading the deduplication data in the cloud storage, the authorized user is only allowed to do so. Based on the output of the data check, the user uploads the file on the cloud. The ratio of health care data after the de-duplication in the cloud storage and the de-duplication of elimination ratio is provided.

$$Elimination\ Ratio\ of\ De-Duplication = \frac{Before\ de-dulication\ in\ MB}{After\ de-duplication\ in\ MB} \tag{7}$$

Fig. 9 shows the storage space for storing de-duplication data in the cloud storage.

Fig. 9 shows that the storage space occupied in the cloud environment requires less space for the proposed work. After applying the de-duplication process, proposed method WCA-BF + ASE requires less storage space for various file sizes. For example, 200 MB requires 10 KB, 400 MB requires 22 KB, 600 MB requires 35 KB, 800 MB requires 38 KB, 1000 MB requires 40 KB of storage space.

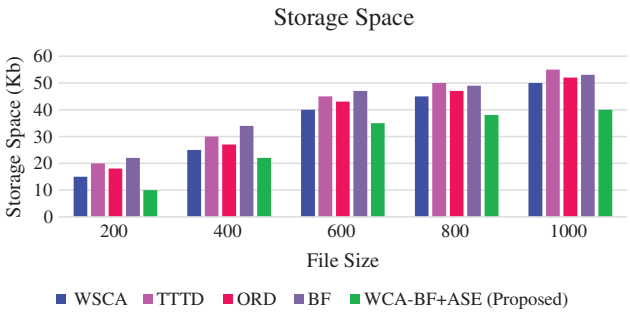

Figure 9: Storage space

5 Conclusion

The contribution of this proposed work is that the data is collected from various sensor devices of the patient, and the collected health care data is considered as a stream that provides chunks based on the window size. The selection of the window size is dependent on the maximum and minimum values of the threshold values. Moreover, the delimiter of the data stream helps to detect the cut-point of the window. Less number of chunks provides better performance in the aspect of storage space for eliminating the redundant values in the health care data by using Bloom Filter, which is applied to the chunks data stream. After processing the de-duplication algorithm of the bloom filter, the reduced data is stored in the FoG server. This work has implemented the concept of an advanced Signature-Based Encryption algorithm. This provides more protection to the health care data. In the study, the security level of WSCA has got 86.5%, TTTD 80%, ORD 84.4%, BF 82%, and the proposed work has got better security storage of 97%. And also, after applying the de-duplication process, the proposed method WCA-BF + ASE has required less storage space for various file sizes. For example, 200 MB has required 10 KB, 400 MB has taken only 22 KB, and 600 MB with 35 KB, 800 MB with 38 KB, 1000 MB with 40 KB of storage space. In the future, it may be extended up to the selection of window size based on the dynamic data and the optimization techniques could be implemented to the health care data.

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

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

References

[1] A. Ullah, K. Hamza, M. Azeem and F. Farha, "Secure healthcare data aggregation and deduplication scheme for FoG-orineted IoT," in *Proc. IEEE Int. Conf. on Smart Internet of Things (SmartIoT)*, India, pp. 314–319, 2019.

[2] M. D. Anto Praveena and B. Bharathi, "An approach to remove duplication records in healthcare dataset based on Mimic Deep Neural Network (MDNN) and Chaotic-Whale Optimization (CWO)," *Concurrent Engineering: Researchand Applications*, vol. 29, no. 1, pp. 58–67, 2021.

[3] H. Zhao, L. Wang, Y. Wang, M. Shu and J. Liu, "Feasibility study on security deduplicationof medical cloud privacy data," *EURASIP Journal on Wireless Communications and Networking*, vol. 1, pp. 1–15, 2018.

[4] A. Sardar and L. E. George, "Data deduplication system based on content defined chunking using bytes pair frequency occurrence," *Symmetry*, vol. 12, no. 11, pp. 1841, 2020.

[5] C. Zhang, D. Qi, Z. Cai, W. Huang, X. Wang *et al.,* "A novel content defined chunkingalgorithm for finding incremental data in data synchronization," *IEEE Access*, vol. 7, pp. 86932–86945, 2019.

[6] M. Marwan, F. AlShahwan, F. Sifou and H. Ouahmane, "Improving the security of cloudbased medical image storage," *Engineering Letters*, vol. 27, no. 1, pp. 1–19. 2019.

[7] H. Ma, Y. Xie and J. Wang, "Revocable attribute based encryption scheme with efficient deduplication for ehealth system," *IEEE Access*, vol. 7, pp. 89205–89217, 2019.

[8] S. Li, C. Xu and Y. Zhang, "CSED: Client side encrypted deduplication scheme based on proofs of ownership forcloud storage," *Journal of Information Security and Applications*, vol. 46, pp. 250–258, 2019.

[9] Y. Yang and X. Zheng, "Privacy preserving smart IoT based healthcare big data storage and self-adaptive accesscontrol system," *Information Sciences*, vol. 479, pp. 567–592, 2019.

[10] L. Xuan, L. Chang and X. Liu, "CE-Dedup: Cost effective convolutional neuralnets training based on image deduplication," arXiv:2109.00899v1, 2021.

[11] L. Chen, F. Xiang and Z. Sun, "Image deduplication based on hashing and clustering in cloud storage," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 4, pp. 1–16. 2021.

[12] M. A. Boura,Q. Lu, F. Zhang, Y. Wan, T. Zhang *et al.,* "Distributed ledger technology for eHealth identity privacy: State of the art and future perspective," *Sensors*, vol. 20, no. 483, pp. 1–16. 2020.

[13] J. Kharat and S. Chougule, "A passive blind forgery detection technique to identify frame duplication attack," *Multimedia Tools and Applications*, vol. 79, no. 11, pp. 8107–8123, 2020.

[14] W. Xia, X. Zou, H. Jiang, Y. Zhou, C. Liu *et al.,* "The design of fast content defined chunking for data deduplication based storage systems," *IEEE Transaction Parallel Distributed System*, vol. 31, pp. 2017–2031, 2020.

[15] R. G. Maqdah, R. G. Tazda, F. Khakbash, M. B. Marfsat and S. A. Asghar, "Content aware deduplication in SSDs," *Journal of Supercomputing*, vol. 76, pp. 8901–8921, 2020.

[16] J. Nie, L. Wu and J. Liang, "Optimization of de-duplication technology based on CDC blocking algorithm," in *Proc. 12th Int. Congress on Image and Signal Processing, Bio-Medical Engineering and Informatics (CISP-BMEI)*, Suzhou, China, 2019.

[17] K. Y. Yigzaw, A. Michala and J. G. Bellika "Secure and scalable deduplication of horizontally partitioned health data for privacy-preserving distributed statistical computation," *BMC Medical Informatics and Decision Making*, vol. 17, no. 1, pp. 1–17. 2017.

[18] W. Sun, X. Chen, X. R. Zhang, G. Z. Dai, P. S. Chang *et al.,* "A multi-feature learning model with enhanced local attention for vehicle re-identification," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3549–3560, 2021.

[19] M. Ali, C. Xu and A. Hussain, "Authorized attribute based encryption multi-keywords search with policy updating," *Journal of New Media*, vol. 2, no. 1, pp. 31–43, 2020.

[20] K. Gu, K. M. Wang and L. L. Yang, "Traceable attribute based signature," *Journal of Information Security and Applications*, vol. 49, pp. 102400, 2019.