Received: April 26, 2021. Revised: May 18, 2021. 

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

# Centralization of Big Data Using Distributed Computing Approach in IoT

**Bishoy Sameeh Sawiris[1]*** **Sayed Abd El-Gaber[2]** **Manal A. Abdel-Fattah[1]**

[1]*Department of Business Information Systems, Faculty of Commerce and Business Administration, Helwan University, Egypt*
* Corresponding author's Email: Bishoy.Sameeh21@commerce.helwan.edu.eg

**Abstract:** Centralized big data is an emerging field that manages information in one common repository. Currently, Internet of Things (IoT) devices are increasing massively for sensing and transmitting data to the big data environment. This information is sensitive and easily traced by attackers due to the centralized data management. To address the security and privacy issues and avoid the system in a single point of failure, in this paper the researchers proposed a new model, namely, BOBS CRABID i.e. Blockchain Based Secure Centralized Big Data model using distributed computing approach. In big data, Hadoop MapReduce is one of the distributed computing approaches which handles all information in a distributed manner. The BOBS CRABID model is designed by three ideas as Multi-Factor authentication, IoT Devices Data Collection and Processing, and Optics based MapReduce for Data Clustering. In multi-factor authentication, a lightweight camellia key generation algorithm (LCKGA) is used for their authenticating all devices based on ID, IP address, MAC and PUF. These credentials are saved in Blockchain Security Entity for authentication verification. All transactions are hashed and stored in the blockchain using Keccak hash algorithm which performs better than the traditional hashing (SHA) 2. Then the researchers conducted IoT devices data collection and pre-processing using Cross Correlation and Min-Max Normalization for redundant data pruning and normalization, respectively. Finally, clustering is performed in a distributed way for optimizing the storage and scalability performance. For that MapReduce is used in which OPTICS algorithm is applied for clustering based on the data size, type of context and nature of data (sensitive or non-sensitive). In this way, data is stored in the big data environment. Experiments were conducted for the proposed BOBS CRABID model and compared with the well-known methods using Hadoop 2.7.2. The proposed BOBS-CRABID model achieves better performance in terms of throughput (increase 1500mbps), response time (reduce 150ms), energy consumption (reduce 35%), attack detection rate (increase 30%), accuracy (increase 20%), computation overhead (300 kb), and time reduce 4s).

**Keywords:** Centralized big data, Internet of things, Distributed computing approach, Mapreduce, Clustering and multi-factor authentication.

## 1. Introduction

Centralized big database is a database which stores and maintains the data in a centralized manner that can be accessed by everyone in the network [1, 2]. It stores and processes data in a large environment and provides data privacy and integrity. Centralized big database is responsible for managing the entire database in the network. Big data consists of large amounts of data and it has three types of data which are structured data, semi structured data and unstructured data. It provides more accurate analysis of data thus leading to more concrete decision-making results [3-5]. Now a days' big data concept is the most suitable business application and IoT application because it processes and analyses huge amounts of data per day. It has both sensitive and non-sensitive data which are stored in centralized big database [6, 7]. A traditional centralized big database has a big problem i.e. security and privacy leakage [8, 9]. Data can be tampered or modified by anyone and that modified data is stored in the database, which increases poor security which is enhanced by deploying blockchain [10-12]. It stores the entire transactions in a hash format which is not tampered by attackers; hence it provides higher security [13,

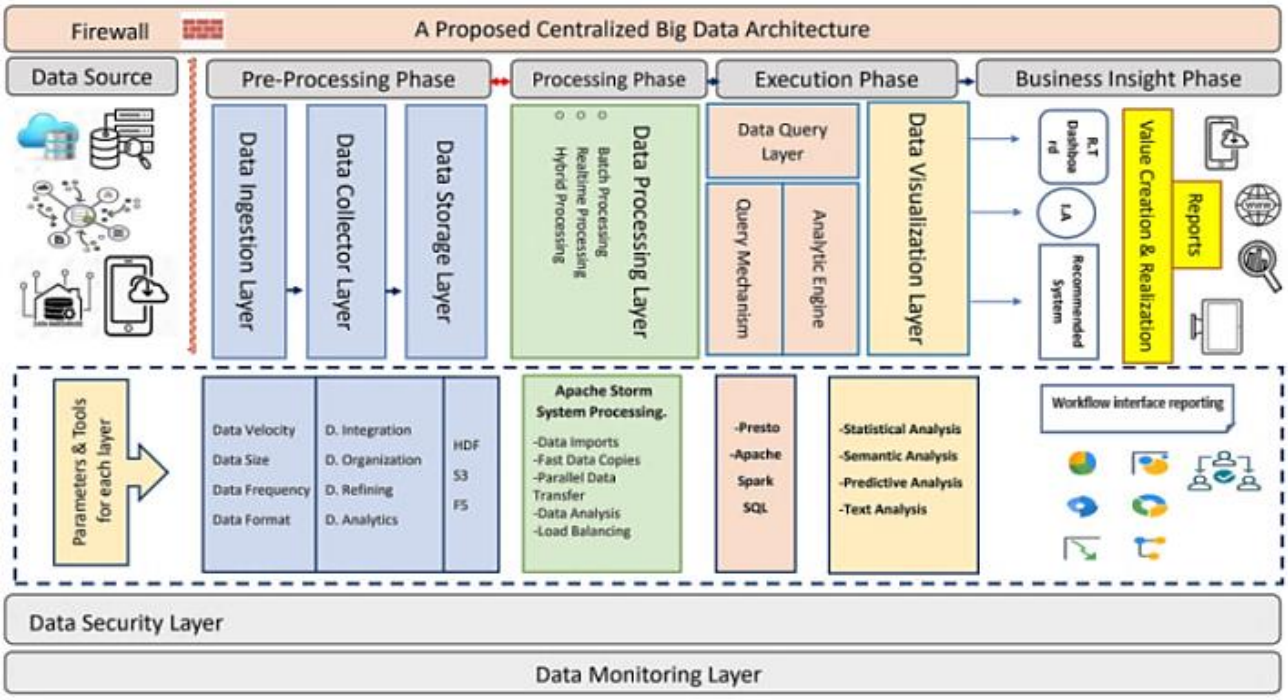


Figure. 1 Architecture of centralized bigdata

14]. However, big data has some challenges such as

- Capturing data
- Storage
- Sharing
- Speed
- Data Diversity

To overcome the challenges of big data, the research introduces Hadoop, which is an open source java-based platform that stores and processes big data. It processes the data in a faster manner by using map-reduce method. It has two functions which are Map and Reduce. In Map Function, the data is split and stored in a distributed server which reduces processing time.

Fig. 1 illustrates the general architecture of centralized big data that includes Hadoop framework, in which the Hadoop storage is enhanced by using Hadoop Distributed File System (HDFS) which divides the data into multiple parts and stores it across clusters of commodity servers as it is a part of the pre-processing phase including the data storage layer. HDFS is able to store different types of data (structured, unstructured, semi –structured) which are shared in a distributed manner. The main aim of this paper is to analyse the data and improve the data storage in the security aspects it is a main focus and important layer in the above architecture is to provide confidentiality, integrity and availability (CIA) to data user. The sub-objectives of this paper as follows;

- To provide security, authentication process is performed for detecting malicious node access to the network.
- To reduce complexity and size of the data, the redundant and similar data are removed from the database.
- To enhance the scalability the data is split into multiple partitions by combination of OPTICS and MapReduce functions.

### 1.1 Research problems in centralized big data

Centralized big data has some challenges such as security threats and resource utilization and storage scalability problems which are defined as follows,

- In security authentication is focused limited and performed without any security credentials, which leads to poor security and some research does not investigate authentication thus increase storage overload.
- All the data are shared without any security credentials leading to poor security. Some research performs hashing, which increase security although the hashing function takes a large number of rounds increasing latency and energy consumption.
- In current research, most of the researchers perform encryption and decryption to enhance security although the cipher algorithm takes much time for encryption and decryption, which increases latency and reduces network performance.

### 1.2 Research contributions

In this paper, the researchers propose a novel model (BOBS CRABID i.e. **B**l**o**ckchain **B**ased **S**ecure **C**ent**ra**lized **Bi**g **D**ata model) for handling transfer of multi-source data from IoT devices to the big data environment. A list of contributions of this paper as follows:

- Firstly, the researchers propose a multi factor authentication for ensuring security. Thus the researchers consider IP, MAC, ID and PUF of the IoT devices which are unique for every device to increase security and prevent malicious node access to the network. In this step, the researchers propose a Lightweight Camellia Algorithm for key generation, which requires a smaller number of rounds, thus reducing latency and increasing network performance. Then, the researchers propose the use of Keccak hash function for performing hash functions, which takes less amount of time to covert hash that reduces latency and energy consumption in comparison with traditional hashing algorithms created in blockchain i.e. $(SHA)^2$.
- Secondly, the researchers propose a Cross Correlation method and Min-Max normalization scheme for reducing redundancy and normalizing data, respectively. In Hadoop, all the redundant data is removed and normalized to increase the storage optimization efficiency and accuracy of data classification or clustering.
- Thirdly, the researchers present MapReduce based clustering which enhances storage scalability by considering data size, type of context and nature of data, thus providing optimal cluster. In this work the researchers used OPTICS based MapReduce method, which increases scalability. In map function all the data are partitioned based on data size, nature and type context and reduce functions construct the cluster which reduce complexity and storage consumption. Finally, all the transactions are hashed and recorded in the blockchain which is not tampered, hence increasing security and network performance.

The performance is analyzed for various evaluation metrics in the Hadoop environment. Finally, it is compared with the previous security schemes to measure the performance of the proposed BOBS CRABID model.

### 1.3 Paper organization

The rest of this paper can be described as follows:

Table 1. Notation table

| Notations | Description |
|---|---|
| $k_l$ | Length of secret key |
| $k_{ll}$ | Left bit of $k_l$ |
| $k_{lr}$ | Right bit of $k_r$ |
| $k_{rl}$ | Left bit of $k_r$ |
| $k_{rr}$ | Right bit of $k_r$ |
| $N_{\epsilon}(d)$ | Neighbourhood radius of point d |
| $R_T$ | Response time |
| $\tau$ | Finishing time |
| n | Beginning time |
| $E_C$ | Energy Consumption |
| $T_E$ | Total Energy |
| $R_E$ | Residual Energy |
| $ADR$ | Attack Detection Rate |
| $\zeta$ | Number of detected attacks |
| $\int$ | Total number of attacks |

Section 2 summarize the literature of centralized database for storing and securing the data in big data and cloud environments. Section 3 highlights the major issues in providing the high security and scalability. Section 4 discusses the proposed BOBS CRABID model in detail, presenting the algorithms procedure. Section 5 presents the simulation results of the proposed BOBS CRABID model which is compared with the previous schemes in terms of different performance metrics. Section 6 includes the conclusion and future work of this study.

## 2. Literature review

In this section, current research works undergone in the centralized big data in cloud environment is discussed in detail.

Authors in [21] design a federated learning for smart home appliances. The proposed system has three main components: customers, manufactures and blockchain. Manufactures are a requested to construct machine leaning for predicting customer product consumption and behaviours. The data are collected from the home appliances by using mobile phones. The features are extracted with adding noise to the features to protect customer privacy. Normalization process is also performed for extracted features to improve the accuracy of leaning. All the transactions are stored in blockchain to enhance security. However, this work depends upon customer training results. It will not suitable for large data because it consumes a lot of time to collect the training results.

A blockchain based trusted data sharing scheme is proposed in [22] for IoT environment which designed by the secure and lightweight triple trusting architecture (SLTA), which includes 2 trusted methods and 3 technologies. The trusted methods are

oracle and DID, which ensures data security and authenticity. The blockchain used asymmetric encryption for providing a private key. DID document mainly focuses on two aspects; encryption and attribution. Combination of digital signature and zero knowledge cryptography method are used for ensuring security. All the transactions are stored in blockchain that is not tampered, thus increasing security and reducing overhead and storage. The personalized big data system is presented as managed by blockchain [23]. In this work, three kinds of blockchain are used private blockchain to store sensitive private information, public blockchain to store public information, and consortium blockchain to store both public and private information which is not tampered by anyone. The proposed method used on chain and out of chain methods is suitable for consortium blockchain, which reduces data amount thus solving storage related problems. A blockchain based data sharing platform [24] used fine grained access control. The proposed system has four functions: system initialization, user registration, data upload and data download. System initialization can run on BSDS-FA platform. HABE algorithm is used to generate both public and main key which are applied to smart contract. In registration phase legal users are registered. In data upload phase, the data are shared in an encrypted format that is stored in smart contract. In data download phase, the user request is verified by smart contract, if it is valid, then the user can access the data. In smart contract the verification algorithm is used for verification process. The result shows that the proposed system achieves high security and reliable data sharing without affecting the performance of data download.

Authors in [25] addressed the problem of public auditing in cloud data. For avoiding the high cost of third-party auditor, this research deploys blockchain technology and it also improves data security and reduces overhead. The proposed system had two entities: data owner (DO) and cloud storage provider (CSP). First, the DO divides the data into blocks is encrypted by using hash function. Then DO constructs merkle hash tree (MHT) by hash tags to verify data integrity. All the hash is stored in the blockchain, and finally the DO updates the encrypted blocks into CSP. The proposed system performs against 51% of attack in the network. A distributed deduplication scheme is presented in [26] for big data in cloud environment. First, the client partitions the data and collects the fingerprints. The proposed Boafft method uses data routing algorithm with respect to similarity, which reduces network overhead and bandwidth. Similarity table is used to avoid duplication of data in the network. Finally, the Boafft method improves data deduplication ratio in a single node using hot fingerprint cache container. The result shows that the proposed system archives high deduplication ratio compared to existing systems. In Boafft method all the data are stored in the data server without any security, hence it is easily tampered by attackers, reducing the network performance. In [27] authors proposed a blockchain based data integrity method for large amount of data in IoT environment. The proposed system includes four entities such as smart contracts, data owner devices, data consumer devices and cloud service providers. Each CSP can provide cloud storage services for data owners. The data integrity is achieved by using bilinear mapping. For verification, the proposed system uses SC- verification algorithm. The experimental result shows that the proposed system achieves high performance in terms of overhead and computational cost for large scale IoT environment. However large scale IoT data has data missing problems hence data recovery is an important process that is not investigated in this paper. In [28] authors addressed the problem of trust and security in IoT environment. For that, this author proposed blockchain based decentralized trust management and secure control mechanism for IoT big data. This research used permissioned blockchain for trust management; first authentication is performed to identify the legitimate user, for that user registers their information into the blockchain. The blockchain provides secret key for legitimate user which is known as certification based key agreement. All the transactions or operations are stored in a cryptography-signed manner in blockchain smart contract, which provides high level security. A blockchain based access control scheme is proposed in [29] big data environment for authorization, revocation. For privacy preservation, a lightweight symmetric encryption algorithm is used. In this paper, two kinds of security attacks are investigated: modifying or stealing data. Here, attacker can modify data by performing specific attack operations as SQL injection, and Identity Hijacking. Other attack behaviour is one that modifies M in the authorization entity. An experimental model is implemented on Aliyun Cloud. For the performance evaluation, computation overhead, throughput and storage overhead are computed. There are two drawbacks in this work: (1) resource constraint issue for mobile terminal devices due to the implementation of double SHA-256, and (2). Chain of records increases computational complexity in storage of health records in the cloud server, which increases complexity on both data owner and user side during matching for retrieval. A multi-cloud environment is

presented for data sharing in decentralized manner [30]. This model is reliable and collaborative for focusing on three types of entities: Data Owners, Miners and Third Parties. A consortium blockchain is used in this paper to guarantee the data privacy and security. In particular, there are five different goals investigated in this paper as follows. (1) Provide data sharing environment for users on the cloud service provider, (2), Obtain resources sharing with trust and consensus, (3), Support the collaboration between the multiple entities, (4), Ensure appropriate access control for all entities and (5), Provide an incentive for accurate and robust data transmission from cloud server to the users. A multi-cloud with blockchain increases complexity, which increases response time for data users. In [31], authors proposed secure environment for data storage, computations and sharing for large scale environment which uses more artificial intelligence (AI) scheme that produces rules for privacy preservation. Various internet connected devices are involved in the network that encounters the users for mitigating DDoS attacks and securing the network by generating and producing the rules. In [32], authors proposed blockchain technology for secure computing in which the Decentralized Virtual Machine Agent model is used exploiting the new mobile agent technology for trust computation and verification. In addition, mobile agent need is to perform reliable data storage and monitoring. In case of data tampering, data user uses warning message for informing to the blockchain and then it updates by 'block and response'. On various aspects, blockchain based secure environment is implemented as Anti-Eavesdropping, Tamper Proof and Anti-discard.

## 3. Problem statement

The major issue in centralized big data is addressing security and privacy issues and low scalability since it does not support or tolerate a large number of users. In [33], cluster-based hybrid data storage is implemented for provenance management. In particular, there are three phases used: Clustering, Hybrid Storage and Feedback. However, provenance node clustering is performed based on its node importance, but it is not sufficient to provide optimal clusters, thus increasing the complexity of data management. Further, hot and cold data scheduling is marinated by LRU queue but generally LRU requires more hardware assistance, which increases complexity. Finally, all the users are considered as legitimate user, which increases processing complexity and reduce security. In addition, data are stored in hybrid storage without any security, which is easily tampered or compromised by the attackers, leading to poor security and reducing the performance of the network.

A decentralized authentication framework is proposed using distributed computing approach. This model is implemented for hospital patients' management and health records are encrypted through SPECK and then stored in blockchain [34]. This paper has several drawbacks, as follows:

- Here, distributed authentication is performed to ensure security, but this research does not tell about the security credentials, if the credentials are not sufficient, which leads to poor security.
- In this proposed system, SPECK algorithm is used for encryption. It requires many rounds during encryption and decryption, which increases complexity and latency.
- Here, SHA-256 hash function is used for performing hashing operation. It takes a big number of rounds to complete hash operation, which increases high latency and energy consumption.

A security privilege-based data sharing is implemented over the cloud environment [35]. Firstly, data is partitioned into multiple parts with respect to the sensitivity type. Based on the similarity, sensitive files are clustered. However, all the information and keys are shared without any security. Hence, it can easily be compromised by the attackers, minimizing the system performance and affecting scalability. Then, the partitioned data are stored in access tree this is suitable for only small size of data. When comes a large amount of data, the access tree has a large amount of child nodes, which increases complexity and storage consumption. In [36], authors proposed ETL based blockchain, which consists of address and transactions. The significant challenges in this paper as follows:

- Here, lambda architecture is used for processing ETL operations, which takes much time to perform clustering and updating thus increasing latency and reducing network performance
- Here blockchain hash function also takes much time to convert the data into hash functions, which increases latency and energy consumption.

## 4. BOBS CRABID model

In the proposed BOBS CRABID model, the researchers focus on the concept of centralization of big data using distributed method that is shown in fig 2. For that, the researchers introduced Hadoop in this system. It is used for storing and processing a large amount of data in a distributed manner. The BOBS CRABID model is designed using three kinds of

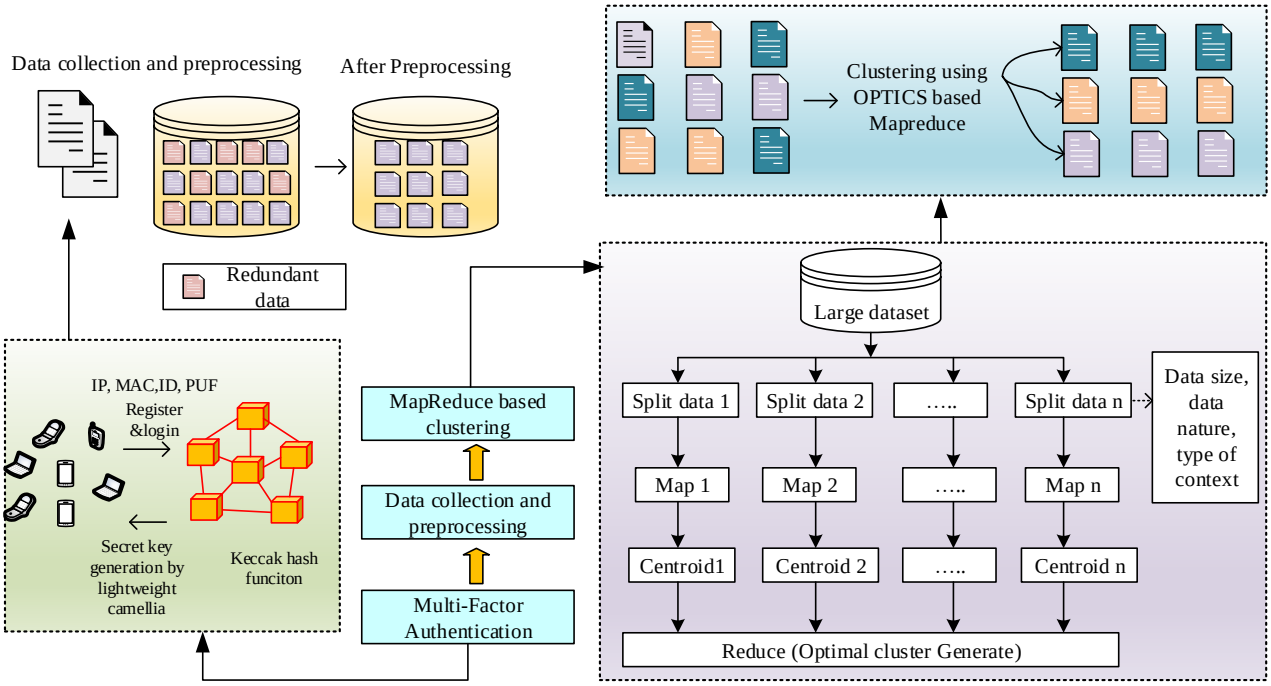


Figure. 2 System architecture

phases, as follows:

- Multi-factor authentication
- Data Collection and pre-processing
- MapReduce based Clustering

### 4.1 Multi-Factor authentication

Initially, the IoT nodes are registered into the blockchain by considering the factors of ID, IP address, MAC and PUF. The blockchain send' a secret key for registered device. In authentication, the blockchain validates whether the IoT device is authenticated or not. Only legitimate nodes can access the network, which helps to predict e malicious node access the network. During authentication, the secret keys are generated by using LCKGA. Due to the resource constraint issues of mobile terminal or IoT devices, Camellia based lightweight block cipher algorithm is used.

In LCKGA, camellia uses 128 bits of block length and key length. A data storage is based on the secret key $sk_i$. The $sk_i$ is shared into two different sub-keys in which each one has 64bits length. Further, the researchers deliberate two different variables with the size of 128-bits and 4-variables with the size of 64 bits, which is defined as follows,

$$K_{ll} = Left - bit\ of\ K_l\ (64) \quad (1)$$

$$K_{lr} = Right - bit\ of\ K_l\ (64) \quad (2)$$

$$K_{rl} = Left\ bit\ of\ K_r\ (64) \quad (3)$$

$$K_{rr} = Right\ bit\ of\ K_r\ (64) \quad (4)$$

Based on the left bits and right bits, the size of key is determined.

- For instance, 128-bits length of key consists of $K_l = K, K_r = 0$
- Similarly, 192-bits length of key consists of $K_l = 128\ left\ bit\ of\ K$
  For 256 bits length of key consists of $K_{rr} = \sim K_{rl}$
  $K_{rl} = 64\ right\ bit\ of\ K$

Finally, the researchers obtained, $K_r = 128\ left\ bit\ of\ K, K_r = 128\ right\ bit\ of\ K$ where, the four 128 bit length created the variables $k_l, k_r, k_a, and\ k_b$ that calculates all the sub keys which have 64 bit length $k_n$, $kw_n$, and $kl_n$. Sub keys are used for encryption and decryption.

All the transactions are hashed and stored in the blockchain. For hashing, the researchers used **Keccak** hash algorithm which performs hashing in a faster manner, thus reducing latency and energy consumption. Keccak is a cryptography hash function that is also known as secure hash algorithm 3. In Keccak, hashes are generated for input data of any size and hash bits are 224, 256, 384, and 512. Further, this algorithm chooses cryptographic sponge in the selected size.

The configurable parameters in Keccak are as follows:

- Data Block Size
- Algorithm State Size

- Number of Rounds in the $f$ function

From the set of configured parameters of Keccak, an optimum set of parameters are selected for improving the cryptographic stability and the performance in applications. The improvement that differentiates Keccak from other hashing algorithms are: eliminated slow modes are as e.g. ($C = 768, C = 1024$), simplified filling algorithm, and introduced functions with an extended outcome for hash messages.

Based on three kinds of operations, which are Initialization, Absorption and Squeezing, in this paper Keccak hash function is implemented. Fig. 3 represents the illustration of Keccak hashing.

A pseudocode for Keccak for hash computation for storing into the blockchain is as follows:

**Keccak – Hash Generation**

KECCAK [$R$,$C$] (Mbytes || $m$ −bits)
{
// **Padding Phase**
$D$ = 2^| $m$ −bits | + sum for i=0..| $m$ −bits -1 of 2^i* $m$ −bits [i]
$p$ = $m$ −bits || d || 0x00 || … || 0x00
$p$ = $p$ XOR (0x00 || … || 0x00 || 0x80)
// **Initialization Phase**
$S[x, y]$ = 0, for $(x, y)$ in (0…4, 0…4)
// **Absorbing Phase**
For each block $Pi$ in P
$S[x, y] = S[x, y]\ xor\ Pi[x + 5 * y]$ For $(x, y)$ such that $x + 5 * y < r/w$
S = KECCAK - $f[R + C]$ $(S)$
// **Squeezing Phase**
Z = empty string
While output is requested
Z = Z ||$S[x, y]$, For $(x, y)$ such that $x + 5 * y < r/w$
S = Keccak-$f[R + C]$(S)
Return Z
}

In pseudocode of Keccak function, D represents the delimited suffix, which performs the trailing bits Mbits and its length. The padded message p is organized to an array of blocks Pi themselves organized as arrays of lane. A variable s holds the state of array lanes. The parallel operator represents the usual string concatenation. In first step, the message m-bits are padded with the sufficient number of zeros for making the message size divisible by the block size of the message. In sponge structure, the padding operation is implemented based on the size of the state in which state size is 1600 bits and 24 rounds. The message blocks are process concurrently for the input data size.

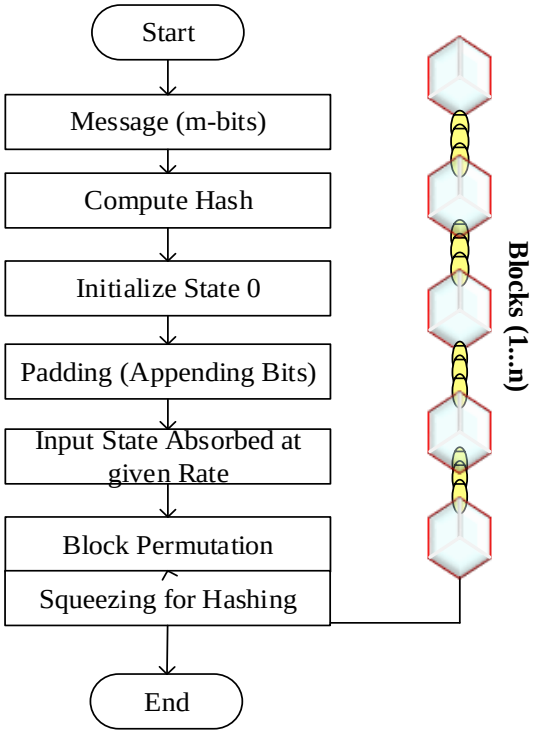


Figure. 3 Keccak hashing

### 4.2 Data collection and preprocessing

After completing authentication, data are collected from the authenticated IoT devices. It has large amount of data (bigdata) that increases complexity. Hence, the researchers perform preprocessing to reduce the size of the data. In preprocessing, the researchers perform two steps: redundant data removal and data normalization. Normally, big data comprises many redundant data items that increase the size of the data that are reduced by considering similarity and correlation factors. For that the researchers deploys Cross Correlation Method, which removes redundant and similar data. For instance, if two data P and Q items are present in the database, then these two data items are said to be similar if and only this satisfies the following condition,

$$\delta(P, Q) = \delta(Q, P) \tag{5}$$

Where, $\delta$ refers to the similarity value. The cross correlation of P and Q can be represented as,

$$cc = \begin{cases} 1, if\ P \leftrightarrow Q\ is\ high \\ -1, if\ P \leftrightarrow Q\ is\ low \end{cases} \tag{6}$$

Where, $\delta(P, Q)$ can be expressed as,

$$\delta(P, Q) = \left(\frac{P \cap Q}{P \cup Q}\right) \times \frac{PR(P,Q)+1}{M - AR(P,Q)+1} \tag{7}$$

Where, $\mathrm{PR}(P,Q)$ denotes the common reading between P and Q and AR $(P,Q)$ denotes the missing reading between P and Q. The intersection and union function can be formulated as,

$$P \cap Q = \sum_{k=1}^{M} Min(W_{Pi}, W_{Qi}) \quad (8)$$

$$P \cup Q = \sum_{k=1}^{M} Max(W_{Pi}, W_{Qi}) \quad (9)$$

If the cc of P and Q is found to be 1 then the removal of redundant data is carried out otherwise the data is are not removed. After completing removal of redundant data, the researchers perform data normalization by Min-Max Normalization algorithm. The normalization of data is carried out to make sure that every feature considered for cross correlation possesses same scale. The formulation of min-max normalization is represented as:

$$x' = \left(\frac{x - minbound\ of\ x}{maxbound\ of\ x - minbound\ of\ x}\right) \times (a-b) + b \quad (10)$$

Where $x$ is the feature and $x'$ is the normalized value of the feature. The $minbound\ of\ x$ denotes minimum value of $x$ and $maxbound\ of\ x$ denotes the maximum value of $x$ respectively and $a\ and\ b$ are predefined boundaries.

### 4.3 Map Reduce based clustering

Map Reduce is a significant process in organizing the large amount of data in an effective manner. It is a design model for processing and generating large data sets with a distributed, parallel algorithm on a cluster. Traditional clustering algorithms are expensive and slow while handling large volume of data and to address these issues in this paper the researchers proposed clustering with MapReduce model. Recent works have used K-means, K-means++, and DBSCAN algorithms which failed in absolute centroid prediction and distance computation based on the number of clusters and density size. In OPTICS clustering algorithm, conventional issues of clustering are addressed. Fig. 4 represents the result of clustering using OPTICS clustering algorithm.

A Map Reduce program comprises Map phase to perform sorting [such as sorting employees by ID into queues, one queue for each ID] and filtering and a Reduce phase that implements an immediate operation such as calculating the number of employees in each queue, yielding name frequencies. The MapReduce process is responsible for calculating scores during processing by arranging the distributed servers, running multiple tasks correspondingly, managing the transfer of data between various parts of the system. In the proposed model the pre-processed data are clustered using OPTICS based Map Reduce Method. The working of both map and reduce phases are provided below:

- Map Phase: The data are classified into multiple partitions based on several metrics by which the data should be partitioned.
- Reduce Phase: The reduce function computes the formation of clusters using OPTICS algorithm. Finally, the data are divided into multiple clusters which reduce data size, complexity, processing time.

In the former phase, the partitioning of data is carried out based on parameters such as data size, type of context, nature of data (sensitive, non-sensitive). The nature of data is classified into three sub-classes based on the sensitivity of the data. The sensitivity of data takes a value between 0 and 1from which the three classes can be derived as follow:

$$Sensitivity\ (\mu) = \begin{cases} low, if\ \mu = (0.1 - 0.29) \\ medium, if\ \mu = (0.3 - 0.69) \\ high, if \mu = (0.7 - 1.0) \end{cases} \quad (11)$$

The formation of clusters is computed by means of OPTICS algorithm, in which the density, rather than centroid formation, of data is considered. The density of data is computed by formulating core distance and the connectivity between the data items is computed by reachability distance. Here, each data is considered as a point and each point is evaluated to be the core point by using:

$$Cdist_{\epsilon,Minp}(d) = \begin{cases} notdefined, if |N_\epsilon(d)| < Min_p \\ Min_p\ small\ distance\ in\ N_\epsilon(d), othwise \end{cases} \quad (12)$$

In this formula, $\epsilon$is the maximum number of points in the database, $N_\epsilon(d)$ is the neighborhood radius of the point d, and $Min_p$ is the minimum points in the $\epsilon$ neighborhood of point d. the search strategy adopted for selecting the neighbor node is termed as greedy method. The reachability of point d to another point e is computed by:

$$Rdist_{\epsilon,Minp}(e,d) = \begin{cases} not\ defined, if |N_\epsilon(d)| < Min_p \\ \max\left(Cdist_{\epsilon,Minp}(d), dist(d,e)\right), otherwise \end{cases} \quad (13)$$

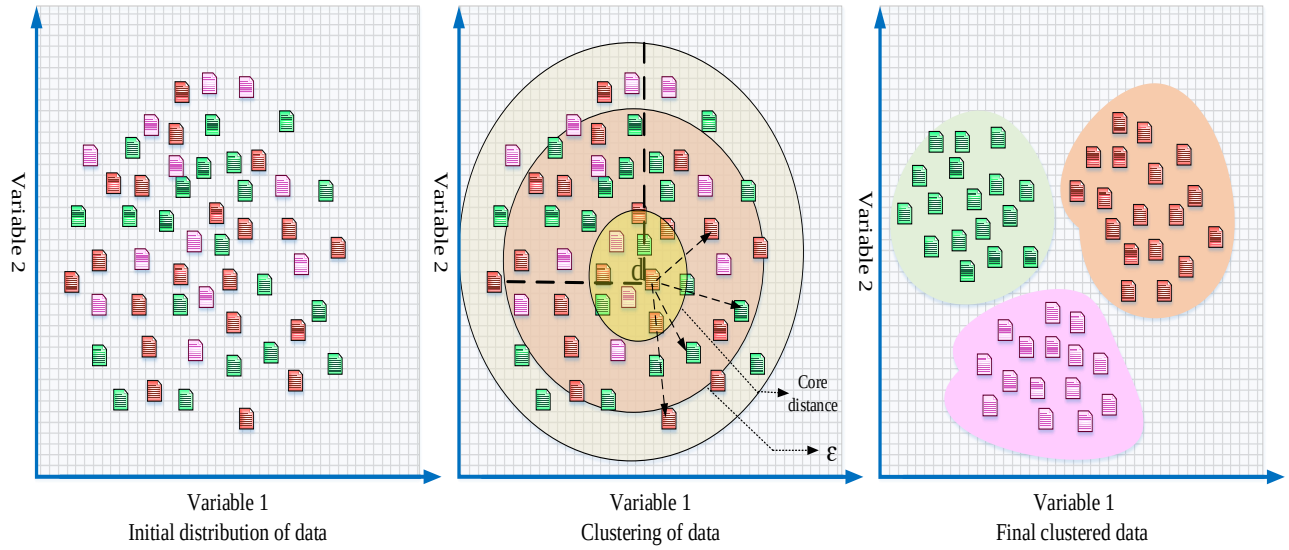


Figure. 4 Optics based clustering

**OPTICS based Map Reduce Algorithm**

*Map phase ();*

Parameters: data size, type of context, nature of data.

If sensitivity ($\mu$) = $0.1 - 0.29$

$\mu$ = low;

Else if sensitivity ($\mu$) = $0.3 - 0.69$

$\mu$ = medium;

Else

$\mu$ = high;

*Reduce phase();*

For each point d do

d. reachability – distance = not defined

For each new point d do

N = search neighbors (d, $\epsilon$)

Set d as treated

If $Cdist$ (d, $\epsilon$, $Min_p$) != not defined then

Queue = empty

Update (N, d, queue, $\epsilon$, $Min_p$)

For each next e do

N' = search neighbors (e, $\epsilon$)

Set e as treated

If $Cdist$ (e, $\epsilon$, $Min_p$) != not defined do

Update (N', e, queue, $\epsilon$, $Min_p$)

End For

End For

End If

Using Eq. (13) two points (d, e) are checked for whether they belong to the same cluster. Similarly, by computing Eq. (13) the multiple clusters of data are constructed. By reducing the value of $\epsilon$, the number of points in the neighborhood gets reduced

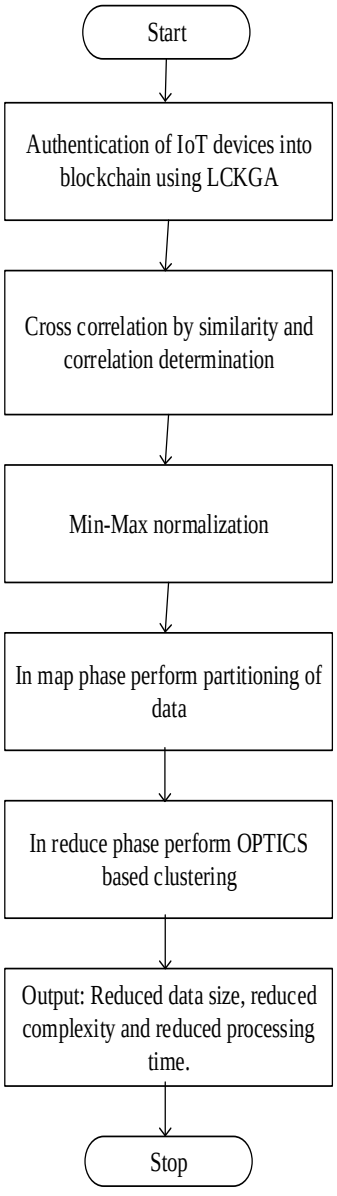


Figure. 5 BOBS-CRABID Flowchart

thereby precisely performing the formation of clusters. The main advantage of the proposed cluster formation technique is that it does not priory require the total number of clusters to be formed. In the 2D, format density of cluster is proportional to the deepness of valley. The run time for formation of multiple clusters of data is derived to be $O(n.\log n)$. The pseudo code for the proposed OPTICS based MapReduce algorithm is described below, in which the clear demonstration of both map phase and reduce phase is provided.

## 5. Experimental study

This section explains the results, which include three sub sections: simulation setup, comparative study and research summary of the proposed BOBS CRABID model. The result section proved that the proposed BOBS CRABID model achieves better performance compared with existing models. In the following, a detailed description of the experiment results is presented.

### 5.1 Simulation setup

This section summarizes the simulation setup of the proposed BOBS CRABID model which is experimented and evaluated in the Hadoop environment. Table 2 illustrates the system specifications.

Fig. 6 represents the Hadoop command that is start-dfs.sh command which means the Hadoop environment starts the name node and data node by using this command before starting MapReduce command.

Fig. 7 represents the start-yarn.sh command in Hadoop environment. All the start-.sh commands are used to start the Hadoop daemons at once which issues the command to the master node to start the daemons on all the nodes of the cluster.

Fig 8 represents the jps command in the Hadoop environment. Jps is a type of command which is implemented for checking all the Hadoop daemons, such as data node, name node, node manager and resource manager that is run on the machine. It is used to verify if a particular daemon is present or not. Fig represents the simulation environment of Hadoop. Hadoop environment has a number of users, one master node and number of data nodes. All the processing is stored in the blockchain to ensure the security of the system. The master nodes include the information of a number of data nodes. Each data node processes multiple jobs. One job includes multiple tasks. The jobs are given by the user to the master node.

Table 2. System & hardware specification

| | | |
|---|---|---|
| System specifications | OS | Ubuntu 14.04.LTS |
| | Apache Hadoop | 2.7.2 |
| | Connectivity | 100Mbps Ethernet LAN |
| | IDE | Netbeans 8.0 |
| | Tool kit | JDK 1.8 |
| | Hard Disk | 1 TB |
| Hardware Specification | Processor | Intel corei7-7700 |
| | CPU core | Quad core |
| | CPU Speed | 3.6GHZ |
| | RAM | 8 GB |

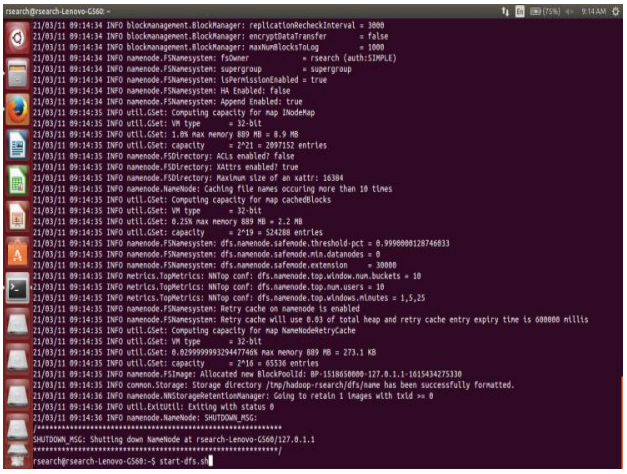

Figure. 6 Start-dfs.sh command in hadoop environment

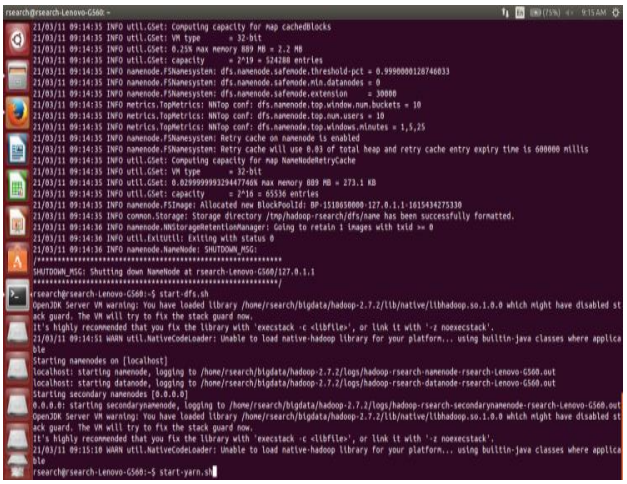

Figure. 7 Start-yarn.sh command in hadoop environment

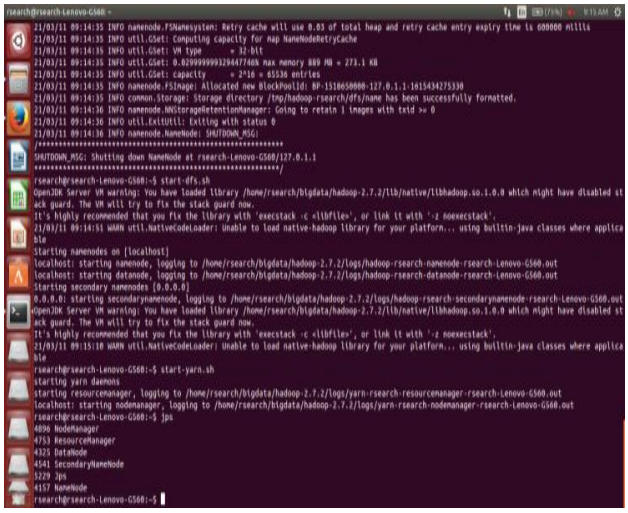

Figure. 8 Jps command in Hadoop environment

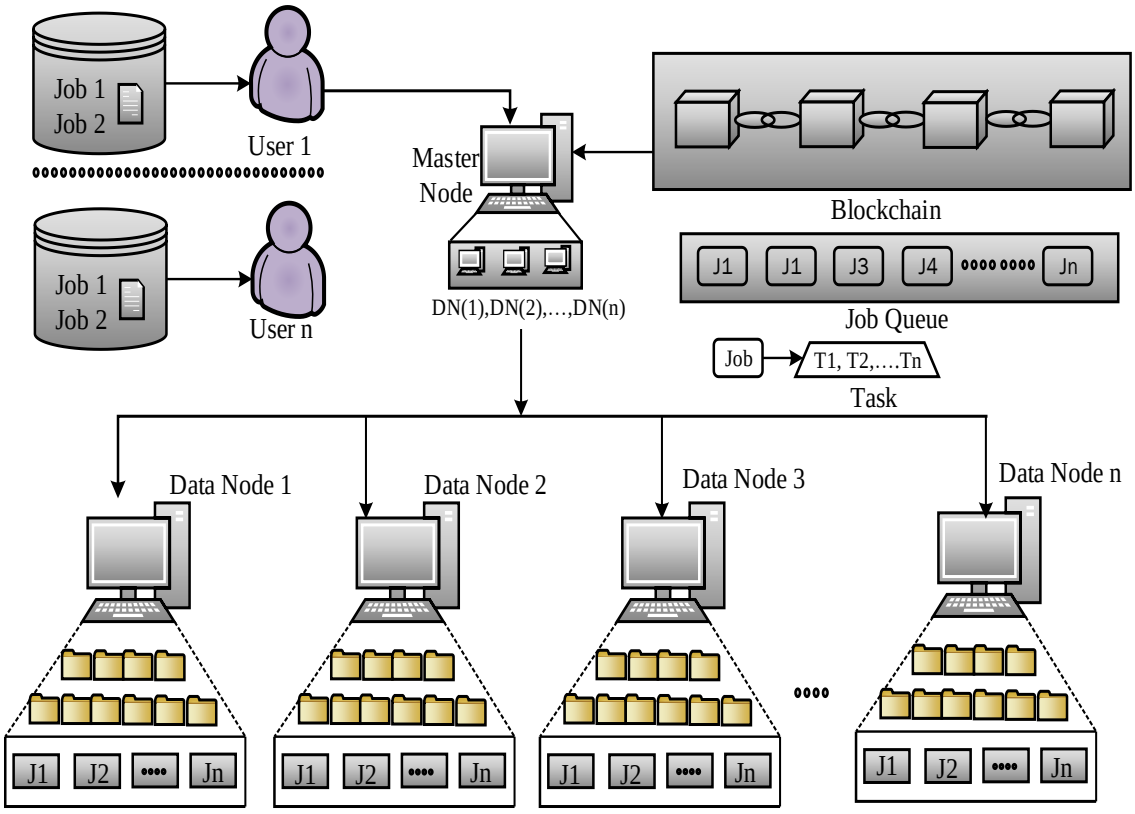


Figure. 9 Experiment environment

## 6. Comparative study

In this section, the researchers illustrate the performance of the proposed BOBS-CRABID model in terms of throughput, response time, energy consumption, attack detection rate, accuracy, computation overhead and time. The performance of BOBS-CRABID is compared with two previous works, DDP [37], and Hybrid BID [33]. The experimental results show that the proposed BOBS-CRABID model achieves better performance when compared with the existing methods.

### 6.1 Impact of throughput

Throughput measures are based on how many data items will be processed in a given amount of time. Throughput enhances the performance of the proposed BOBS-CRABID model. Fig 10 represents the comparison of throughput for the proposed BOBS-CRABID model and existing model which is

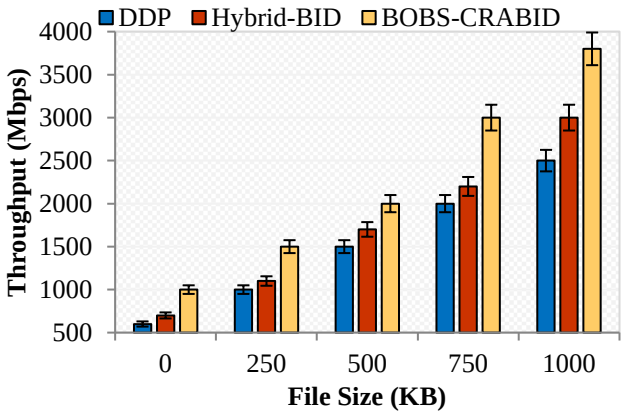


Figure. 10 Throughput vs. file size

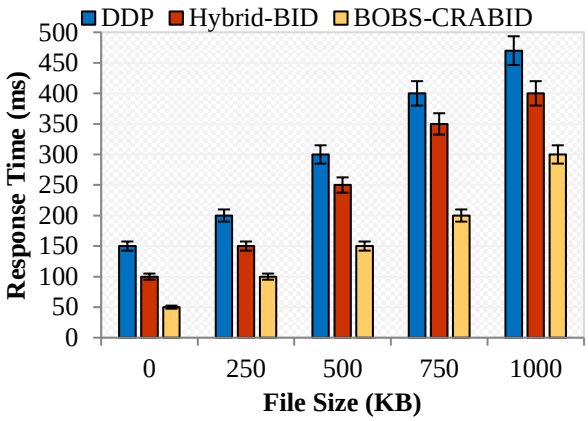


Figure. 11 Response time vs. file size

evaluated with respect to file size. The comparison results show that the proposed BOBS-CRABID achieves high throughput compared with existing models.

The BOBS-CRABID model achieves high throughput because the researchers perform MapReduce based clustering method, in which the data are divided into multiple clusters which reduces complexity and processing time and increases throughput. The BOBS-CRABID method achieves 1000 mbps higher than Hybrid-BID model and 1500 mbps higher than DDP method.

### 6.2 Impact of response time

Response time represent the total amount of time taken by the network for processing the data that are evaluated with respect to file size. In other words, response time calculates the difference between the finishing time and beginning time that is defined as follows:

$$R_T = \tau - n \tag{14}$$

In this Formula, $R_T$ represents the response time and $\tau$ represent the finishing time and $n$ represent the beginning time. Fig 11 illustrates the comparison of proposed and existing response time with respect to file size. Fig 10 clearly states that the proposed BOBS-CRABID model achieves low response time compared with the existing models such as DDP and hybrid BID because the researchers proposed cross correlation method and min max normalization algorithm for removing redundant data and similar data respectively, thus reducing data size, which leads to low response time when performed a large amount of data. Thus, this work achieves less response time compared existing works. BOBS-CRABID model achieves 100ms less than Hybrid-BID method and 150ms less than DDP method.

### 6.3 Energy consumption

Energy consumption denotes the total energy consumed by the environment during data processing, which is calculated by the difference between the total energy and residual energy that is defined as follows:

$$EC = T_E - R_E \tag{15}$$

In this Formula, $EC$ represents the energy consumption and $T_E$ is the total energy and $R_E$represent the residual energy.

Fig. 12 represents the comparison of proposed BOBS-CRABID model and existing model energy consumption with respect to number of transactions. The result shows that the proposed BOBS-CRABID model consumes less energy compared with other existing models, like DDP and hybrid BID. In this work, the researchers perform IoT node authentication that allow only authorized users to access the environment, which prevents the malicious users from entering the environment, thus reducing energy consumption because of collecting the information from the authorized user instead of collecting it from all users (authorized and unauthorized). The researchers additionally divided the data into multiple clusters thus reducing data size and complexity. This also reduced energy consumption compared with existing works. As compared to the previous works, the proposed

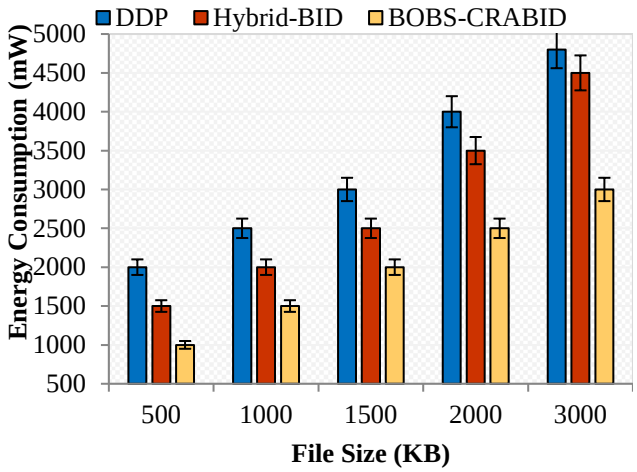


Figure. 12 Energy consumption vs. file size

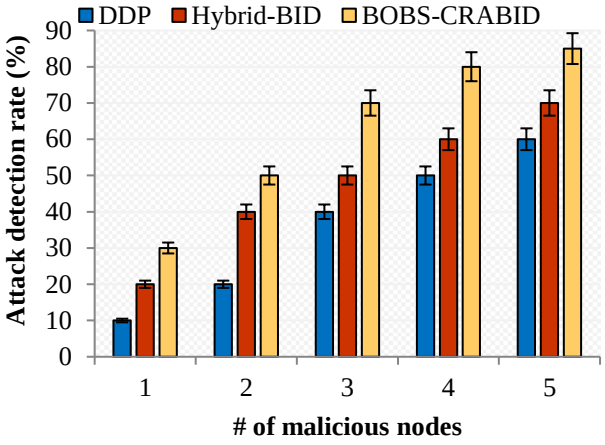


Figure. 13 Attack detection rate vs. no. of malicious nodes

BOBS-CRABID model reduces energy consumption by 35%, which improves the lifetime of the network.

### 6.4 Impact of attack detection rate

Attack detection rate is defined as the number of attacks detected from the total amount of attacks in a particular amount of time, which is calculated with respect to the count of malicious nodes, which is defined as follows:

$$ADR = \frac{\zeta}{\int} \times 100\% \qquad (16)$$

In this Formula, Where ADR represent the attack detection rate and ζ represent the number of detected attacks and ∫ represent the total number of attacks.

Fig 13 represents the comparisons of proposed and existing model attack detection rate with respect to number of malicious nodes. Fig 13 clearly states that the proposed BOBS-CRABID model achieves high attack detection rate compared to existing works. In this work the researchers perform IoT node authentication, thus avoiding unauthorized node entering the environment, which increase attack detection rate. All the transactions are stored in the blockchain which is not tampered by anyone and also the proposed method performs against unauthorized users. By doing this, the proposed model achieves a high attack detection rate compared with existing methods. The proposed BOBS-CRABID model achieves high attack detection rate, 30% higher than DDP method and 20% higher than Hybrid-BID method.

### 6.5 Impact of accuracy

Accuracy is used to measure the correctness of the proposed BOBS-CRABID model. Accuracy is calculated by the addition of true positive and true negative that is divided by all the samples.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (17)$$

In this Formula, TP represents true positive and TN represents TN, and FP is a false positive and FN is a false negative.

Fig 14 represents the comparison of three models, including proposed model. Accuracy is evaluated with respect to number of malicious nodes. Accuracy is increased exponentially with the increasing number of malicious nodes. The comparison results show that the proposed method achieves high accuracy compared with others. In this work, the researchers perform authentication to detect and prevent malicious nodes, thus increasing the proposed model accuracy. Redundant and similar

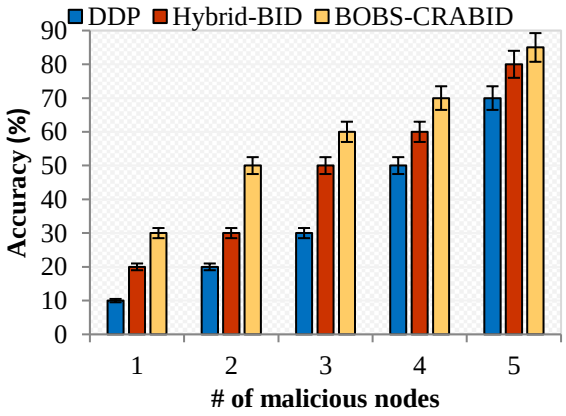


Figure. 14 Accuracy vs. no. of malicious nodes

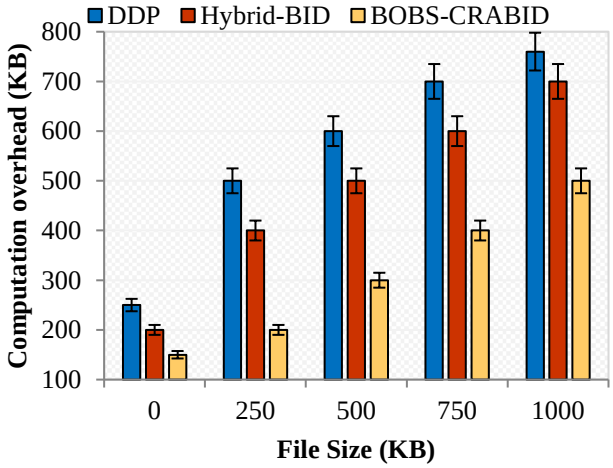


Figure. 15 Computation overhead vs. file size

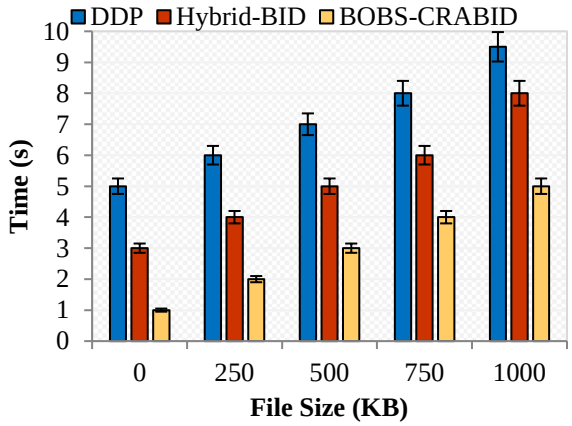


Figure. 16 Time vs. file size

data are removed from the data thus reducing data size and increasing accuracy. BOBS-CRABID achieves accuracy in 10% higher than Hybrid-BID and achieves 20% higher than DDP method.

### 6.6 Computation overhead

Computational overhead is represented as the extra load which restrict the reliability of the Hadoop environment, that occur due to the big number of requests sent by the IoT nodes in a particular time, which reduces overall performance of the network.

Fig. 15 represents the comparison of proposed and existing models computation overhead with respect to file size. The result shows that the proposed BOBS-CRABID model achieves less computation overhead compared with other existing models. In this work, authentication is performed to avoid unauthorized network thus reduces computation overhead. The researchers performed MapReduce based clustering, in which the data are clustered by OPTICS method, thus reducing complexity of the process. Hence, the researchers achieved low computation overhead compared with DDP and hybrid BID model. BOBS- CRABID models reduces 300 kb computation overhead compared with Hybrid-BID model and achieves 200 kb computation overhead compared to DDP model.

### 6.7 Computation overhead

This metric calculates the overall time taken by BOBS-CRABID model during data processing. It represents the efficiency of the proposed model.

Fig 16 represents the comparison of time for both proposed and existing model. The result shows that the proposed BOBS-CRABID model consumes less time to process the given data. The time increases exponentially with the increasing file size. In this work, cross correlation and min max normalization algorithm is used for removing redundant and similar data, thus reducing size of the data, which reduces processing time. The data are divided into multiple clusters, which reduce complexity and overall processing time. Therefore, the researchers consumed less time to process the data compared with existing models. The proposed BOBS-CRABID model consumes 3s less than Hybrid-BID model and 4s less than DDP model.

### 6.8 Simulation setup

This section summarizes how the proposed BOBS-CRABID model has improved the superior performance compared to existing models. Fig (-) explains the performance of the proposed BOBS-CRABID model in terms of throughput, response time, energy computation, attack detection rate, accuracy, computation overhead and time with the user of BOBS-CRABID model. The major research highlights are discussed as follows:

- For enhancing security, the researchers performed multi factor authentication. For that, it deploys lightweight camellia key generation algorithm, which helps to prevent malicious node access the network.
- For accurate data analysis, the researchers performed two types of pre-processing, such as redundant data removal and data normalization, which reduces processing complexity and size of the data.

Table 3. Comparison of proposed and existing works

| Performance Metrics | Proposed vs. Existing Approaches | | |
|---|---|---|---|
| | DDP | Hybrid BID | BOBS-CRABID |
| Throughput (Mbps) | 1520 | 1740 | 2260 |
| Response time (ms) | 304 | 250 | 160 |
| Energy Consumption (mW) | 3260 | 2800 | 2000 |
| Attack detection rate (%) | 36 | 48 | 63 |
| Accuracy (%) | 36 | 48 | 59 |
| Computation overhead (KB) | 480 | 350 | 100 |
| Time (s) | 7.1 | 5.2 | 3 |

- For enhancing storage scalability, the researchers performed OPTICS based clustering by considering type of context, nature of data and data size, thus increasing scalability and reducing processing time.

Table 3 summarizes the numerical analysis of the proposed BOBS-CRABID model which illustrates the average values of the performance metrics. In Addition, the table represents the comparison of proposed BOBS-CRABID model and existing model.

## 7. Conclusion and future work

In big data assisted IoT environment, security, privacy and scalability are the significant issues. One of the important challenges in big data environment is centralized storage, which causes single point of failure and high computational overhead in data processing, storage and retrieval. To overwhelm these issues, the researchers proposed BOBS CRABID model which utilizes distributed computing approach for handling large volumes of data from multiple sources of IoT devices. Firstly, IoT devices are authenticated using multiple factors as ID, Password, IP address, MAC address and PUF. For that, LCKGA algorithm is used, which generates secret key and it is validated during authentication. In this way, the researchers increase the security of IoT devices. Then, hashing is applied to provide integrity features for all data transmitted from the IoT devices. This information is stored in the blockchain which is not tampered by attackers. Hence the researcher's method provides high security Secondly, all devices sensed events are collected and transmitted to the blockchain. Data gathered from IoT devices, are processed using cross correlation and min-max normalization methods. This step reduces redundant data and improves the efficiency of storage optimization and improves the accuracy of data classification and clustering. Thirdly, MapReduce based OPTICS algorithm in which clusters are formed and then stored to the Hadoop environment is used which reduces the complexity and storage consumption. An evaluation of the proposed BOBS CRABID model is experimented and compared with the previous works. From the experiments, it is concluded that the performance of the proposed work has shown better results in terms of throughput (increase 1500mbps), response time (reduce 150ms), energy consumption (reduce 35%), attack detection rate (increase 30%), accuracy (increase 20%), computation overhead (300 kb), and time reduce 4s).. In the future, the researchers have planned to use deep reinforcement algorithms for key update and revocate operations in the blockchain based big data environment.

## Conflicts of Interest

The authors declare no conflict of interest.

## Author Contributions

Abstract, Introduction, Research Methodology, Literature Review, Proposed Model, Comparative Study, Performance Analysis, Results, and Conclusions were conducted by Bishoy Sameeh & Manal Abdel Fattah and Sayed Abd el-Gaber.

## Acknowledgments

This work was not supported by any organization.

## References

[1] M. Babar, F. Khan, W. Iqbal, A. Yahya, F. Arif, Z. Tan, and J. Chuma, "A Secured Data Management Scheme for Smart Societies in Industrial Internet of Things Environment", *IEEE Access*, Vol. 6, pp. 43088-43099, 2018.

[2] S. Shi, D. He, L. Li, N. Kumar, M. Khan, and K. R. Choo, "Applications of blockchain in ensuring the security and privacy of electronic health record systems: A survey", *Computers & Security*, Vol. 97, 101966, 2020.

[3] N. Elisa, L. Yang, F. Chao, and Y. Cao, "A framework of blockchain-based secure and privacy-preserving E-government system", *Wireless Networks*, Vol. 1, pp. 1-11, 2020.

[4] M. Z. Bhuiyan, A. Zaman, T. Wang, G. Wang, H. Tao, and M. Hassan, "Blockchain and Big

Data to Transform the Healthcare", In: *Proc. of the International Conference on Data Processing and Applications*, pp. 62-68, 2018.

[5] J. Chi, Y. Li, J. Huang, J. Liu, Y. Jin, C. Chen, and T. Qiu, "A secure and efficient data sharing scheme based on blockchain in industrial Internet of Things", *Journal of Network and Computer Applications*, Vol. 167:102710, 2020.

[6] J. Li, W. Jigang, G. Jiang, and T. Srikanthan, "Blockchain-based public auditing for big data in cloud storage", *Information Processing and Management*, Vol. 57, No. 6, 2020.

[7] Z. Zhang, L. Huang, R. Tang, T. Peng, L. Guo, and X. Xiang, "Industrial Blockchain of Things: A Solution for Trustless Industrial Data Sharing and Beyond", In: *Proc. of IEEE 16th International Conference on Automation Science and Engineering (CASE)*, pp. 1187-1192, 2020.

[8] J. Huang, Y. W. Qi, M. R. Asghar, A. Meads, and Y. Tu.-C, "MedBloc: A Blockchain-Based Secure EHR System for Sharing and Accessing Medical Data", In: *Proc. of 18th IEEE International Conference on Trust, Security and Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 2019.

[9] Z. Li, H. Guo, W. M. Wang, Y. Guan, A. Vatankhah, G. Q. Huang, and X. Chen, "A Blockchain and AutoML Approach for Open and Automated Customer Service", *IEEE Transactions on Industrial Informatics*, Vol. 1, 2019.

[10] O. J. A. Pinno, A. R. A. Gregio, and L. C. E. De Bona, "ControlChain: Blockchain as a Central Enabler for Access Control Authorizations in the IoT", In: *Proc. of GLOBECOM - IEEE Global Communications Conference*, pp. 1-6, 2017.

[11] M. Usman and U. Qamar, "Secure Electronic Medical Records Storage and Sharing Using Blockchain Technology", *Procedia Computer Science*, Vol. 174, pp. 321-327, 2020.

[12] D. Berdik, S. Otoum, N. Schmidt, D. Porter, and Y. Jararweh, "A Survey on Blockchain for Information Systems Management and Security", *Information Processing & Management*, Vol. 58, 102397, 2021.

[13] K. M. Khan, J. Arshad, and M. M. Khan, "Investigating performance constraints for blockchain based secure e-voting system", *Future Generation Computer Systems*, pp. 13-26, 2019.

[14] I. Butun, P. Osterberg, and H. Song, "Security of the Internet of Things: Vulnerabilities, Attacks and Countermeasures", *IEEE Communications Surveys & Tutorials*, pp. 616-644, 2019.

[15] M. Naisuty, A. Hidayanto, N. C. Harahap, A. Rosyiq, A. Suhanto, and G. M. Hartono, "Data protection on Hadoop distributed file system by using encryption algorithms: A systematic literature review", *Journal of Physics: Conference Series*, Vol. 1444, 2020.

[16] Y. Xu, S. Wu, M. Wang, and Y. Zou, "Design and implementation of distributed RSA algorithm based on Hadoop", *Journal of Ambient Intelligence and Humanized Computing*, Vol. 11, pp. 1047-1053, 2020.

[17] Y. Su, G. Shen, X. Sun, and Z. Tang, "Realization of Chaotic Sequence Encryption Algorithm in MapReduce Distributed Parallel Model", In: *Proc. of IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 1650-1655, 2018.

[18] U. Narayanan, V. Paul, and S. Joseph, "A light weight encryption over big data in information stockpiling on cloud", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 17, pp. 389-397, 2020.

[19] P. Camacho, B. Cabral, and J. Bernardino, "Insider Attacks in a Non-secure Hadoop Environment", *Recent Advances in Information Systems and Technologies. WorldCIST, Advances in Intelligent Systems and Computing*, Vol. 570, 2017.

[20] P. Paul and D. Veeraiah, "Multi-Layered Security Model for Hadoop Environment: Security Model for Hadoop", *International Journal of Handheld Computing Research*, Vol. 8, pp. 58-71, 2017.

[21] Y. Zhao, J. Zhao, L. Jiang, R. Tan, D. Niyato, Z. Li, L. Lyu, and Y. Liu, "Privacy-Preserving

Blockchain-Based Federated Learning for IoT Devices", *IEEE Internet of Things Journal*, Vol. 8, pp. 1817-1829, 2021.

[22] S. Peichang, W. Huai-min, S. Yang, C. Chang, and Y. Wen-tao, "Blockchain‐based trusted data sharing among trusted stakeholders in IoT", *Software - Practice and Experience*, pp.1-14, 2019.

[23] J. Chen, Z. Lv, and H. Song, "Design of personnel big data management system based on blockchain", *Future Generation Computer Systems*, Vol. 101, pp. 1122-1129, 2019.

[24] H. Xu, Q. He, X. Li, B. Jiang, and K. Qin, "BDSS-FA: A Blockchain-Based Data Security Sharing Platform with Fine-Grained Access Control", *IEEE Access*, Vol. 8, pp. 87552-87561, 2020.

[25] J. Li, W. Jigang, G. Jiang, and T. Srikanthan, "Blockchain-based public auditing for big data in cloud storage", *Information Processing & Management*, Vol. 57, 2020.

[26] S. Luo, G. Zhang, C. Wu, S. Khan, and K. Li, "Boafft: Distributed Deduplication for Big Data Storage in the Cloud", *IEEE Transactions on Cloud Computing*, Vol. 8, pp. 1199-1211, 2020.

[27] H. Wang and J. Zhang, "Blockchain Based Data Integrity Verification for Large-Scale IoT Data", *IEEE Access*, Vol. 7, pp. 164996-165006, 2019.

[28] M. Zhao-feng, W. Ling-yun, W. Xiao-chang, W. Zhen, and Z. Weizhe, "Blockchain-Enabled Decentralized Trust Management and Secure Usage Control of IoT Big Data", *IEEE Internet of Things Journal*, Vol 7, pp. 4000-4015, 2020.

[29] L. Tan, N. Shi, C. Yang, and K. Yu, "A Blockchain-Based Access Control Framework for Cyber-Physical-Social System Big Data", *IEEE Access*, Vol. 8, pp. 77215-77226, 2020.

[30] M. Shen, J. Duan, L. Zhu, J. Zhang, X. Du, and M. Guizani, "Blockchain-Based Incentives for Secure and Collaborative Data Sharing in Multiple Clouds", *IEEE Journal on Selected Areas in Communications*, Vol. 38, pp. 1229-1241, 2020.

[31] K. Wang, J. Dong, Y. Wang, and H. Yin, "Securing Data with Blockchain and AI", *IEEE Access*, Vol. 7, pp. 77981-77989, 2019.

[32] P. Wei, D. Wang, Y. Zhao, S. K. Tyagi, and N. Kumar, "Blockchain data-based cloud data integrity protection mechanism", *Future Generation Computer Systems*, Vol. 102, pp. 902-911, 2020.

[33] D. Hu, D. Feng, Y. Xie, G. Xu, X. Gu, and D. Long, "Efficient Provenance Management via Clustering and Hybrid Storage in Big Data Environments", *IEEE Transactions on Big Data*, Vol. 6, pp. 792-803, 2020.

[34] A. Yazdinejad, G. Srivastava, R. Parizi, A. Dehghantanha, K.R Choo, and M. Aledhari, "Decentralized Authentication of Distributed Patients in Hospital Networks Using Blockchain", *IEEE Journal of Biomedical and Health Informatics*, Vol. 24, pp. 2146-2156, 2020.

[35] E. Zaghloul, K. Zhou, and J. Ren, "P-MOD: Secure Privilege-Based Multilevel Organizational Data-Sharing in Cloud Computing", *IEEE Transactions on Big Data*, Vol. 6, pp. 804-815, 2020.

[36] R. Galici, L. Ordile, M. Marchesi, A. Pinna, and R. Tonelli, "Applying the ETL Process to Blockchain Data. Prospect and Findings", *Information (Switzerland)*, Vol. 11, 2020.

[37] B. Sawiris and M. Abdel-fattah, "A Novel Solution for Distributed Database Problems", *International Journal of Advanced Computer Science and Applications*, Vol. 11, 2020.