IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 6, 2018, accepted December 9, 2018, date of publication December 28, 2018, date of current version January 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2890099

An Integrated Methodology for Big Data Classification and Security for Improving Cloud Systems Data Mobility

ISMAIL HABABEH[1], (Member, IEEE), AMMAR GHARAIBEH[1], (Member, IEEE), SAMER NOFAL[1], (Member, IEEE), AND ISSA KHALIL[2], (Member, IEEE)

[1]School of Electrical Engineering and Information Technology, German Jordanian University, Amman 11180, Jordan
[2]Qatar Computing Research Institute, Doha, Qatar

Corresponding author: Ismail Hababeh (ismail.hababeh@gju.edu.jo)

ABSTRACT The expand trend of cloud data mobility led to malicious data threats that necessitate using data protection techniques. Most cloud system applications contain valuable and confidential data, such as personal, trade, or health information. Threats on such data may put the cloud systems that hold these data at high risk. However, traditional security solutions are not capable of handling the security of big data mobility. The current security mechanisms are insufficient for big data due to their shortage of determining the data that should be protected or due to their intractable time complexity. Therefore, the demand for securing mobile big data has been increasing rapidly to avoid any potential risks. This paper proposes an integrated methodology to classify and secure big data before executing data mobility, duplication, and analysis. The necessity of securing big data mobility is determined by classifying the data according to the risk impact level of their contents into two categories; confidential and public. Based on the classification category, the impact of data security is studied and substantiated on the confidential data in the scope of Hadoop Distributed File System. It is revealed that the proposed approach can significantly improve the cloud systems data mobility.

INDEX TERMS Big data classification, data security, metadata, risk impact level, HDFS.

I. INTRODUCTION

The concept of big data refers to the huge amount of information that the organizations process, analyze, and store [1]. The escalated use of information resources and the need of advanced data processing technologies lead to the existence of big data. An overview of big data variety, volume, security and privacy are discussed in [2].

Big data analysis offers service tools, such as Hadoop Distributed File System (HDFS) [3], [4] which supports managing, storing huge amount of data, fast automated decisions, and decreases the risks of human estimations. The HDFS is accepted as the most widely used dataset tool that supports redundancy, reliability, scalability, parallel processing, distributed architecture systems [5], and designed to handle different big data types; structured, semi-structured and unstructured. Moreover, Hadoop MapReduce Job-Scheduling algorithm [6] supports clustering big data in a spread network environment [7]. In addition, big data analysis provides significant opportunities for solving different information security problems by using Hadoop technologies and HDFS tools [8]. The data value that is generated from big data through the analysis phase is of extreme important [9].

However, the intensive use of big data raises new data security challenges [10]–[13], particularly when processing confidential data, such as, organization's trading secrets, personal and health information. In order to reach the ultimate benefit from this information, we aim to protect confidential data from potential risks.

The importance of cyber security in the presence of big data is discussed in [14] and [15], where big data is considered as a target for hackers to attack high valued information. However, the traditional security solutions are not capable for protecting big data mobility. Thus, securing mobile big data is a challenge that needs new technologies to protect such massive data.

The Fog computing [16] supports a control over the confidential data. The big data transmission between the clouds and the fog imposes a determination of storage devices data type that need some clustering techniques that sort the data in accordance to its heat and discriminate big data based on

the latency sensitivity and data temperature. However, it is necessary to employ security techniques to prevent any data threat or potential risks.

Recently, many researchers have focused on data security. However, most of the security mechanisms are designed to protect fixed data against threats, which in turn are insufficient for big data and beyond the processing capabilities of existing databases [17], [18]. The confidential data is a precious target for threats that negatively affect the organization's trust and credit. For example, big data may create a security threats to user's emails by generating sites for phishing, based on their emails behavior and interest [19]. Big data security is considered one of the most challenges that threaten cloud computing systems and affects the organizations' reputation [20].

Security and privacy are the critical requirements for storing, managing, analyzing and transmitting big data [21]. Any comprehensive big data security solution should meet data confidentiality, integrity and availability.

Big data privacy concerns are considered in building big data environments and protecting big data during either storage or processing [22], [23], where multiple encryption techniques are implemented to prohibit unofficial users from accessing big data [24]–[26]. However, the proposed approaches treat all data with the same priority and their intractable time complexities prove their impracticality.

Big data security is viewed from two complementary aspects particularly, data security and access control [27], where the main big data problems are data management and data classification [28]. Big data security management might be controlled by Kerberos management policy that secures data at deferent levels; communications, transmissions, authorization and storage [11]. The Kerberos is designed for data authentication, transport secure layer for communication, and data encryption. No matter how Kerberos secure data, the big data comes from many sources that might have different security and governance policies, thus make the suggested solutions hard to implement.

Big data security practices are suggested throughout functions such as Know, Prevent, Detect, Respond, and Recover [29] to detect successful attacks so that any data leakage can be identified and countered. Nevertheless, any potential risks such as breaches and attacks on sensitive information are still a challenge.

A big data security prototype that presents access control features of logging system is introduced in [30]. The access control features are utilized when creating big data files.[1] However, the main stump is the difficulty of determining the user behaviors and the overlapping of user roles over the big data. In addition, a method of choosing big data fields that should be secured is proposed in [31], where big data is considered as one object that has its own attributes which are ranked according to their importance. But still no ranking policy can fit the attributes of unstructured big data.

Big data security on transmission over cloud systems can be achieved by secure socket layer SSL connection that take place between the sender and receiver clouds name nodes [32]. A hash value is used to create series of encrypted tickets that are used in the communication between the sender and receiver nodes. However, problems with one certificate authority can affect the user authentication at both cloud sides which allow the attacker to capture the tickets.

State of the art data mining techniques preserve data privacy by substituting the value of sensitive attributes and forbidding the disclosure of sensitive and private data. However, these techniques cannot fit big data and so cannot address its security and privacy concerns [9]. In addition, data mining faced a challenge when addressing unstructured data types [31].

Transmitting big data at different confidential levels from one node to another in the cloud system might expose sensitive and critical data to threats. For example, an invader tampers the data when it is being exchanged by exploiting the system and thus gaining access to critical information. Therefore, big data classification followed by data security technique guarantees safe data mobility in a cloud system while maintaining the performance standards. Thus, minimizes the risks of data aggregation, and analysis of vast amount of confidential data.

In this paper, we aim to overcome the big data security challenges and enhance cloud systems data mobility by reducing potential security threats during data transmission or exchange. Our proposed approach focuses on elevating big data privacy and security, especially when transferring huge amount of sensitive and critical data between cloud systems nodes. The proposed approach presents an integrated methodology for big data classification and security based on the data criticality and sensitivity. We implement an encryption and decryption technique on confidential data before considering any data transmission, duplication and analysis.

The main contributions of this work are:

- Develop a new big data classification technique based on the data contents and the description attributes (metadata) that determine the level of data sensitivity or criticality. This technique resolved the challenges associated with the big data volume by adapting the open source distributed software platform HDFS formatter to perform high speed data classification.
- Classify big data based on predefined policies set by data owner which determine the confidentiality level of the available data and categorizes the files into two groups; public which does not need security and confidential which needs security. This in turn decides the only data that need to be secured and thus reduces the security costs for all transmitted files between cloud nodes.
- Propose a new encryption technique for confidential files to prevent threats that might lead to leakage or loss

[1]Unless otherwise stated, we will use the term ''file'' to represent ''big data file''

of critical data, taking into consideration the efficiency and time consuming.
- Develop a user-friendly software tool to perform big data classification, security, and assist cloud systems administrators in measuring big data mobility performance.
- Integrate big data classification and security into one methodology to accomplish ultimate cloud system throughput in terms of data confidentiality, integrity and availability.

The remainder of the paper is organized as follows: Section II presents the related works of big data analysis and security. The security concerns and potential risks of big data are debated in Section III. The integrated methodology of big data classification and security is discussed in Section IV. The experimental results and performance evaluation are presented in Section V. Finally, Conclusions and future work are presented in Section VI.

II. RELATED WORK

The emergence of big data directs the researcher's attention to the challenges of cloud systems security. The essential concepts of big data architecture and the data risk assessment are discussed [1]. However, the mechanism of applying security controls and information governance on big data are not addressed.

Protecting cloud systems from potential risks is a major challenge for the cloud service providers [33]. Two types of attacks are identified, insider and outsider. The implementation of insider attacks is considered as the bottleneck of cloud security. A cybersecurity framework is proposed to advocate insider attacks in cloud systems. In this framework architecture, the Hidden Markov model technology is used to detect the future behavior of the edge devices that are categorized into four groups according to their effect on the cloud security: legitimate, sensitive, under-attack, and hacked. This framework transfers the edge devices to the virtual honeypot device which is a new technology that identifies malicious edge device where the attacker activities can be followed, predicted, and prevented in the future. Though, the prevention controls are neither introduced nor described.

Preventing potential risks by using encryption tools like Security Information and Event Management SIEM [34] and Data Loss Prevention DLP [35] to protect sensitive data are introduced in the cloud security method [10]. Nevertheless, no settings for critical data indicators are introduced, nor detailed processes of big data classification are provided which make it hard to implement.

Phishing is an important security threat on big data [19]. A case study is used to clarify how phishers can attack big data by creating fake emails to gain their valuable information. However, no phishing protection technique is applied against fake emails.

Big data protection is presented in deferent levels that includes; communications, processing, authentication and storage. Big data integrity and access control approach that protects data during processing and storage is presented in [22]. Researchers in [11] described how to enforce a data protection management policy on huge amount of data while not affecting the performance. Nevertheless, the large time complexity required by the protection technique makes it inapplicable. In addition, as big data resources are varies and controlled by different policies, the suggested data protection solutions are hard to implement.

A security hardening methodology [27] is proposed using attribute relation graph that focused on the value of data and how to extract valuable information from data rather than data protection. The Computing on Masked Data CMD tool is used to combine RND and DET cryptographic methods in order to protect sensitive data. CMD techniques are designed to improve the cloud system confidentiality, integrity and availability. However, CMD techniques failed to cover the overhead during the masking process that requires more time, and the keys management requires efficient solutions.

A method of selecting attributes that should be protected to secure big data is discussed in [31]. Big data is considered as a single object which has its own attributes that are ranked according to their importance. Nevertheless, no ranking policy can fit the attributes of unstructured big data.

The complexity of controlling and maintaining big data in diverse fields of science is described in [2]. The Map Reduce High Performance Computer Cluster HPCC, the Knowledge Data Discovery, and data mining methods are used to define data sets and extract the required information directly. Anyhow, data mining techniques cannot fit big data.

A set of big data security characteristics are defined in [17]. Some automated tools are used to collect and stabilize different data types, and analyze engines to process large degree of fast changing data in real time applications for security analysis. However, there is no methodology introduced that overcomes security threats and privacy violations and there is no description on how to protect big data from different resources against potential risks.

Data confidentiality is considered critical [21] to achieve data privacy which has additional requirements that could be enhanced by encryption and hid data access control techniques. Nevertheless, the difficulty of implementing confidentiality techniques such as access control and encryption make it infeasible for big data.

An intelligent-driven security model to monitor users that possess abnormal behaviors is discussed in [30]. This model is used to enhance security by including dynamic and self-adaptive doubtful user log system, and self-assuring software containing a package that includes the suspicious behaviors keywords inserted by users. When the fishy user exceeds a maximum limit of abnormal behaviors, then he/she is highlighted as a critical user. A model analysis is followed to decide if the behavior is a doubtful action or just out of nosiness. Moreover, the self-assuring framework consists of library to identify and store keywords of the user abnormal behaviors and their critical logs. Though, normal behavior is not defined and there is no protection against data loss and data leakage.

The information security challenges are discussed [20], where new threats detection algorithms are proposed for processing significant amounts of data from different sources, high performance cryptography, data provenance, security visualization, and skilled personnel. Essential concepts are addressed about information security threats and described the SIEM system. However, the proposed methodologies and tools were not be able to achieve big data protection and cannot comply with information security best practices.

The big data study [9] divided big data security into access control, and information security. The proposed security hardening methodology used attribute relation graph, and focused on the data value rather than the data itself. Nevertheless, the relationship between the protected attributes in the selection algorithm is neither defined nor described.

The security of big data analytics techniques are used to generate applicable intelligence and significant meanings from data streams in real-time applications that becomes strong needs for cybersecurity setups [14]. The proposed security analytics technique is tested on local network-related data sets to decrease the false positive rate of a predictive model for fraudulent transactions. Though, local network-related data sets are not enough for testing the predictive model over big data that is generated from multiple sources with different data types.

The privacy and security risks of Hadoop Eco system is presented in [24]. As the Name Node and the Data Node have the entire control over the data, it is essential to implement multiple encryption techniques to prevent unauthorized people from gaining access to the data. Anyhow, this security mechanism costs large time complexity which makes it inapplicable.

The proposed integrated classification and security method in this paper is distinguished by its capability to address the confidentiality level of the file contents. Indeed, our proposed method incorporates a data security algorithm that is specifically designed to minimize the risks of aggregation and migration of huge amount of confidential information.

III. SECURITY CONCERNS AND POTENTIAL RISKS OF BIG DATA

To avoid any potential risks, protecting valuable information of big data is a central goal. Data security is one of the most crucial issues in big data analysis [36] that are proved to be NP-Hard [31]. The main aspects that should be considered in big data analysis are confidentiality, integrity and availability. Confidentiality aims to protect big data by prevent accessing the data except for the authorized users according to data sensitivity. Integrity intents to allow authorized users to modify, edit, update and delete data. Availability guarantees that data is available and accessible.

In many organizations, storing, collecting and processing a huge amount of sensitive information are done in one place. Storing huge amount of confidential data, such as customers' and patients' personal data, trading and financial information, in one place may expose to potential risks; sabotage, data leakage, data loss and hacking. Moreover, it might expose to denial of service due to malicious attack. Therefore, we propose a new risk assessment classification technique that prevents potential threats on big data and promotes risk management based on the following risk metrics values: asset, vulnerability exposure, threat level and likelihood of threat [37]–[39].

Each risk metric is measured as a value between 0 and 5 according to the risk assessment criteria defined in ISO27005:2011 [40], where negligible (0-1), low (1-2), medium (2-3), high (3-4) and very high (4-5).

Threat and Vulnerability value *THRV* is defined as the sum of the threat and the vulnerability values and computed in Equation (1):

$$THRV = Threat + Vulnerability \tag{1}$$

The Risk Impact Level value (*RIL*) determines the security control level required to protect big data. The *RIL* value is based on risk metrics values; the security control level is realized as critical that needs protection, or public where data are opened to the public and protection is not needed. The *RIL* value is defined as the multiplication values of *Asset*, *THRV*, and Likelihood of threat *LTHR* and computed in Equation (2):

$$RIL = Asset \times THRV \times LTHR \tag{2}$$

According to definition of Equation (2), the *RIL* risk level value can be measured as follows: the highest risk level value is evaluated to very high (4-5) when the risk components reach its critical values (4-5), therefore, $RIL = (veryhigh) \times (veryhigh) \times (veryhigh) = (4-5) \times (4-5) \times (4-5)$, where the lowest risk level value is evaluated to negligible (0-1) when the risk components values are in the range of (0-1), hence, $RIL = Negligible \times Negligible \times Negligible = (0-1) \times (0-1) \times (0-1)$. Consequently, the values of *RIL* are inflating to upper risk level values (1-2), (2-3), (3-4), and (4-5) according to the combination of *RIL* components risk level values. Based on the resulted *RIL* value (0-5), the proposed classification method decides the proper security control level that needed to protect critical data during processing, copying and moving over the cloud.

IV. THE INTEGRATED CLASSIFICATION AND SECURITY METHODOLOGY

We propose an integrated methodology which consists of two techniques that classify and secure files in order to achieve high level of security on data mobility in cloud systems. Figure 1 describes the architecture of the proposed method.

The details of the data classification and security architecture are discussed in the following subsections.

A. BIG DATA CLASSIFICATION TECHNIQUE

Basically, big data is classified according to its needs, priorities, and degrees of protection based on data sensitivity and criticality. The proposed classification technique classifies big data into two main categories; confidential and public in

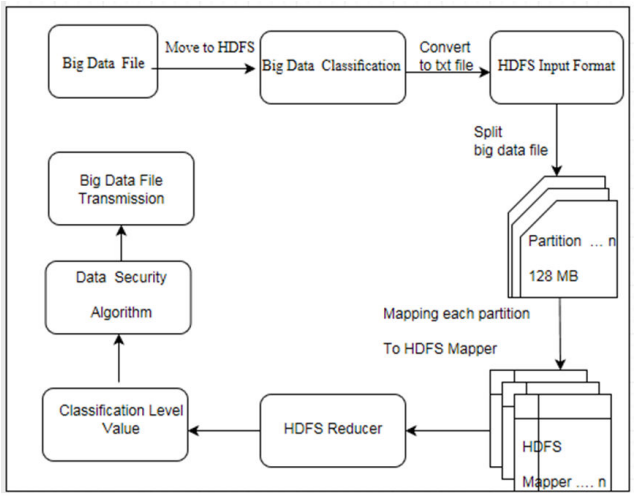

FIGURE 1. Big data classification and security architecture.

accordance to the risk impact level value that is defined as follows:

- Let *RIL* represent risk impact level value that is based on ISO27005:2011 risk assessment metrics values (0-5).
- Let *MAV* represent metadata extended attribute, then *MAV* is computed in Equation (3):

$$MAV = \begin{cases} 0, & 0 \leq RIL \leq 1 \\ 1, & RIL > 1 \end{cases} \quad (3)$$

The *MAV* is evaluated to True (1: confidential) if the risk impact level value between low to very high (1-5) and evaluated to False (0: public) if the risk impact level value is none or negligible (0-1). The *MAV* is inserted in the file metadata upon creation to facilitate classification processes. Our proposed classification algorithm is also applied on the files already created without *MAV*. The description of files categories are detailed as follows:

- Confidential Data: Confidential data represents high sensitive information that can be viewed or accessed by only authorized users as it contains important information related to the organizations and their customers such as financial data, personal, and patient's health information. Any expose or leakage to critical data might cause direct loss or negative impact on the organizations reputation.
  The confidential metadata attributes are filing system features that are not interpreted by the system, but provide additional information about the files like file permissions or modification [41]. In addition, metadata attributes allow system administrators to associate additional metadata for a file or directory. The confidential metadata attributes are implemented to prevent potential risks on files that might contain sensitive information.
- Public Data: Public data is defined as the normal data that includes general information which can be viewed by any user without any restriction on accessing these data files. Therefore, the information in such data files should be opened for the public and should not be protected.

Files exist in different types such as (txt, doc, xml, csv, xls, sql, log, db, pdf, image, audio, video, etc.). The proposed classification technique aims to convert files of type txt, doc, xml, csv, xls, sql, log, and db, into text files and split it in different partitions by using HDFS input formatter [42] that validates the input specification of the task and split the input data file into logical splits, each of size 128 MB to ensure that the reduced splits do not exceed the available memory, or do not need additional techniques to free up the memory. Each split is then assigned to a single Mapper.

Based on the security search value in all partitions, the file is identified as confidential or public. The other file types like pdf, image, audio, and video which could not be converted to text, are classified by their contents during the file creation and their *MAV* are set to confidential or public. However, if such files are already created without metadata attributes, then their classification level should be determined and their *MAV* is inserted accordingly. The big data classification workflow processes are described as follows:

1) Files are uploaded to HDFS for processing.
2) If the file *MAV* is known as true, then the file is determined as confidential and the security algorithm will be applied accordingly. Otherwise, if the file's *MAV* is false, then the file is determined as public which does not need to be secured.
3) If the file's *MAV* is not known and the file is of type pdf, image, audio, and video, or any type that could not be converted to text, then the file's classification level needs to be determined and its *MAV* needs to be inserted.
4) If the file's *MAV* is not known and the file is of type txt, doc, xml, csv, xls, sql, log, db, or any type that could be converted to text, then the HDFS input formatter is used to convert the file contents into text.
5) The converted big data text file is split into smaller input partitions, each of 128 MB using HDFS Customized Input Format (HCIF)
6) Each input partition is assigned to one HDFS Mapper that reads the partition contents and classifies it to confidential or public according to a predefined policy. Each mapper works separately on its data partition.
7) The Mappers outputs are fed into the HDFS Reducer, which combines the mappers classification results and output a single classification level value (0: public, 1: confidential).

B. BIG DATA SECURITY ON DATA MOBILITY TECHNIQUE

The big data security procedures are applied on data mobility between different cloud nodes based on the data file classification level. If the classification level is public, then no security actions are required. Otherwise, the following data security technique is applied:

1) The user sends his metadata to the sender and the receiver clouds that are encrypted by sharable credentials between the sender/receiver cloud and the user.

2) The sender transmits the encrypted data nodes addresses and data blocks IDs to the receiver using random access key.
3) The receiver creates encrypted block access key that is shared with its data nodes.
4) The receiver data node uses the encrypted block access key to trigger a request for copying or moving the data stored at the sender data node.
5) The request is received by the sender data node and decrypted for authentication.
6) Upon passed authentication, the sender data node transmits the data packet to the receiver data node waiting for its response to confirm data packet reception. The sender repeats transmitting data packet until a delivery confirmation is received, or the failed trials are exceeded the allowed limit and the data admin in the sender part is informed.
7) The receiver data node gets the data packet and checks its hash value.
8) If the hash value of data packet is successfully confirmed, then the receiver data node sends encrypted acknowledgment to the sender data node. If for any reason the acknowledgment is missed, the receiver data node may receive multiple copies of the same data packet, in this case, all repeated data packets are ignored.
9) The sender data node receives the acknowledgement and responds to the receiver move data request by deleting the transmitted data or keeping the data if a successful copy is delivered. Figure 2 presents the big data security technique processes

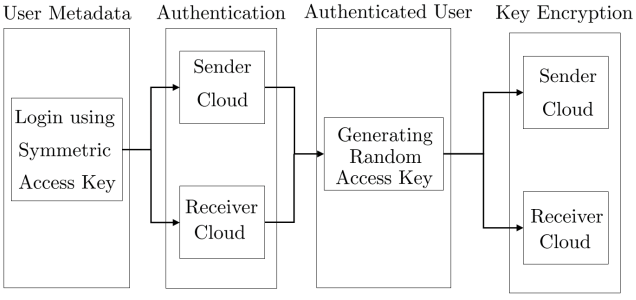

FIGURE 2. Big data security processes.

Analysis of Big Data Security on Data Mobility: The data threats possibilities are investigated and the security procedures are considered carefully for each case. The metadata that transferred between the sender and receiver might be captured by the invader. However, the metadata is encrypted and cannot be decrypted, and hence the invaders attacks are failed even if they present themselves as senders or receivers. The block access keys and the hash value are encrypted/decrypted only by the senders and receivers, while the data transferred is kept in encrypted mode.

A possible threat scenario that could take place is that the invader breaks or holds the transmitted data packets.

Algorithm 1 Big Data Classification

Input: Metadata-attribute value *MAV* of the file
Output: File Classification; Confidential/Public

1: Get the file *MAV* by using the command line: {hadoop fs -getfattr [-R] -n name | -d [-e en] <path>}
2: **if** the file *MAV* exists **then**
3: Go to Algorithm 2
4: **else**
5: **if** the file type is text or any type that could be converted to text **then**
6: Do steps 12-15
7: **else**
8: Do steps 9-11
9: Define the file/folder *MAV*
10: Set the *MAV* by using the command line: {hadoop fs -setfattr -n name [-v value] | -x name <path>}
11: Go to Algorithm 2
12: Convert the file into text partitions by using HCIF
13: Assign one HDFS Mapper to each text partition
14: Use the HDFS Mapper to classify each partition into confidential/public
15: Collect all Mappers' classification results
16: Use HDFS Reducer to reduce all Mappers' classification results into one result
17: **if** one or more Mappers' result is confidential **then**
18: The Reducer result is confidential
19: **else**
20: The Reducer result is public
21: **if** HDFS Reducer result is confidential **then**
22: set confidential to the file *MAV*
23: Go to Algorithm 2
24: **else**
25: set public to the file *MAV*
26: Go to Algorithm 2

In this case, the data availability is checked by a confirmation of data packet delivery and the hash value. Thus, the data integrity is achieved.

C. THE INTEGRATED CLASSIFICATION AND SECURITY ALGORITHMS

The proposed big data methodology is modeled by two algorithms; big data classification presented in Algorithm 1 and security on classified data mobility presented in Algorithm 2. The notations used in both algorithms are summarized in Table 1.

V. EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION

Files are different in their nature; some have structured data, other have semi-structured data, and the rest have unstructured data. In addition, big data might contain some information that should kept open to the public. Therefore, we developed a Map-Reduce framework based on our

Algorithm 2 Big Data Security on Data Mobility

Input: UMD, BAT, T_{scd}, TO, $MaxRet$, N_{rt}, N_{dc}
Output: Secured file

1: **if** the file MAV is confidential **then**
2: Do steps 5-22
3: **else**
4: Go to step 22
5: SCN sends $\{\text{UMD}\}_{K_t}$ to RCN
6: RCN shares BATs with RCD
7: RCD sends $\{\text{BATS}\}_{K_t}$ to SCD and requests DT
8: SCD decrypts BATs and verifies the request authenticity
9: SCD sends $\{\text{DT}\}_{K_t}$, Hash$\{\{\text{DT}\}_{K_t}\}$ to RCD, starts T_{scd}
10: RCD gets $\{\text{DT}\}_{K_t}$, Hash$\{\{\text{DT}\}_{K_t}\}$, verifies the Hash
11: RCD sends $\{\text{acknowledgment}\}_{K_t}$ to SCD
12: **if** $T_{scd} < TO$ **then**
13: SCD waits for the acknowledgment
14: **else**
15: **if** $N_{rt} < MaxRet$ **then**
16: Repeat step 9
17: **else**
18: The administrator in SCD is prompted
19: **if** $N_{dc} > MaxRet$ **then**
20: The administrator in RCD is prompted
21: SCD receives the acknowledgment and deletes DT from its part
22: End

TABLE 1. Classification and security algorithm abbreviations.

Abbreviation	Definition
MAV	Metadata-Attribute Value
HDFS	Hadoop Distributed File System
HCIF	HDFS Customized Input Format
HDFSM	HDFS Mapper
HDFSR	HDFS Reducer
TXT	file type that could be converted to text
SCN	Sender Cloud Name Node
SCD	Sender Cloud Data Node
RCN	Receiver Cloud Name Node
RCD	Receiver Cloud Data Node
K_t	Encryption Key
UMD	User Metadat
BAT	Block Access Token
DT	Data to be transfered
T_{scd}	Timer of SCD
TO	Timeout Value
$MaxRet$	Maximum number of retransmissions
N_{rt}	Number of SCD retransmissions
N_{dc}	Number of RCD received duplicate copies

previous work [43], [44] to analyze the results of the proposed techniques and to validate its performance. In this framework, we tested the feasibility of the proposed classification and security algorithms. Moreover, the proposed framework supports secured data mobility decision making in addition to the use of knowledge extraction.

The decision tree is one of the key issues in secured big data classification. As the development of cloud real-time applications boosts, the volume of such applications' data is inflated. For example, classifying huge amount of data to identify the enterprise sensitive data that needs to be secured is a complex task. Therefore, a parallel distributed decision tree technique; Hadoop Map-Reduce framework is used to accomplish this challenge task. A decision tree in this case consists of one root node and multiple decision nodes. A measure function is applied to determine the preferable splitting security attribute that is used by HDFS input formatter function to split the big data into multiple data partitions. In each partition, if all data belongs to a specific security class, the input formatter function terminates and the decision tree is generated. Otherwise, the input formatter function continues its splitting process recursively until all data partitions belong to the same security class, or no splitting attribute is left, then the decision tree is generated accordingly. Consequently, an appropriate security classification rules are applied to classify new data of undistinguished security class labels.

A. EXPERIMENTAL SETUP

The big data classification and security techniques are tested on different data file types of sizes approximately 1, 2, 4, 8, and 16 GBs. Apache Hadoop 2.6.0 is installed on PowerEdge R720 servers, 2 processor sockets - 6 Core Intel Xeon E5-2630, 64 GB RAM, runs OS Linux Ubuntu 10.04, java 1.0.7- openjdk, PuTTY 0.70 authentication system [45] and connected with 1GB Ethernet NICs.

B. EXPERIMENTAL RESULTS

Several experiments were setup to evaluate the performance of classifying and securing big data CSV files in terms of classification time, response time, throughput, and delay time.

TABLE 2. Files classification time (sec).

file Type	1 GB	2 GB	4 GB	8 GB	16 GB
CSV	93.6	187.1	374.2	748.4	1496.9
SQL	96.9	195.7	389.0	794.7	1524.8
LOG	93.2	182.2	368.8	742.4	1491.5
XLS	94.0	190.5	380.1	758.9	1516.0

1) BIG DATA CLASSIFICATION TIME

Table 2 summarizes the generated classification time required to classify 5 different file sizes of types CSV, SQL, LOG, and XLS into public/secured.

2) BIG DATA TRANSMISSION RESPONSE TIME

Big data transmission between clouds is allowed after user authentication. User authentication at the sender and the receiver clouds are carried out by integrating a user private key with a public key. Our security experimental results are compared with non-secured Hadoop baseline [4] and the secured methods proposed in [46] and [32].

The approach in [4] describes the design of HDFS which is used for big data storage and computation tasks through large number of nodes. The huge amount of aggregated data enabled by HDFS can raise several challenges

about data security, availability and consistency. In addition, the current HDFS does not support intra-cloud data encryption which makes data security becomes a challenge issue. Moreover, built-in HDFS security features such as Kerberos and Access Control Lists which are not adequate to be used only for enforcing role-based access to sensitive data. Therefore, an appropriate security controls are needed to achieve big data security needs.

The technique discussed in [32] introduces the security data migration between clouds storage that are based on HDFS. The security processes are initiated from the source cloud to a target cloud based on the users' demand. The user authentication is validated in both source and target clouds. The source cloud initiates a secure socket layer SSL connection between the name nodes of both source and target clouds. The target name node generates a temporary session key, a random number and series of tickets encrypted by the session key to communicate with the source nodes, compute the double hash value and return the encrypted tickets to the source name node respectively.

The method proposed in [46] presents a hybrid encryption method to protect file blocks and session keys based on HDFS. Symmetric encryption technique is adopted to encrypt and decrypt file blocks at clouds data nodes while asymmetric encryption technique is used to secure the symmetric keys. This method prevents intruders from holding data from data nodes and ensuring that users are light weighted. However, the proposed method introduces high overhead performance while the architecture overhead is negligible.

TABLE 3. Big data transmission response time (sec).

Method	1 GB	2 GB	4 GB	8 GB	16 GB
HDFS Baseline [4]	125.1	255.6	549.2	1143.1	2383.4
Secured HDFS [46]	169.7	378.9	801.0	1739.7	3689.0
Secured HDFS [32]	158.3	344.4	736.3	1588.4	3351.1
Proposed Secured HDFS	141.5	297.8	627.4	1332.5	2825.3

Table 3 presents the average response time (SDTRT) required for securing and transmitting 5 different CSV files using secured and non-secured HDFS methods at rate of 64Mb/s. The data mobility process requires that the receiver cloud should know the necessary metadata at the sender cloud name node. Our protocol achieves this requirement by using a common key that established by the user between the sender and the receiver clouds. This key serves as an authentication to both clouds that enable them to generate and verify the necessary tokens without holding them for transferring data from one cloud to another. Therefore, our approach decreases the unnecessary network bandwidth and processing overhead caused by other approaches in comparison [32] and [46] due to extra transmission, encryption, and decryption operations between the sender and the receiver clouds. Hence, significance decreases in the total transmission response time, the total delay time, and the throughput degradation are achieved by our protocol which results in better HDFS security performance.

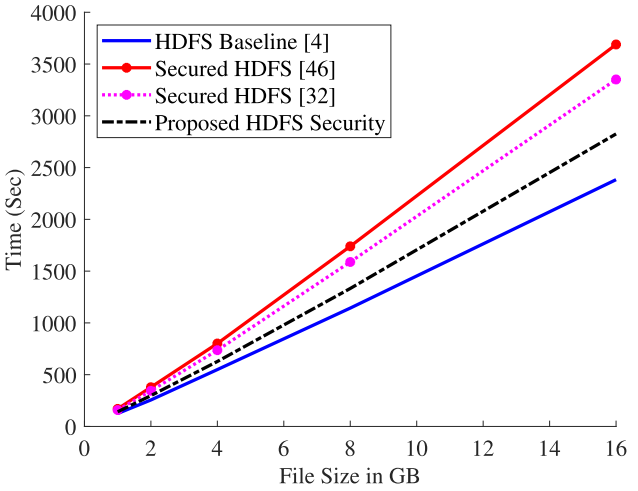

FIGURE 3. Big data transmission response time.

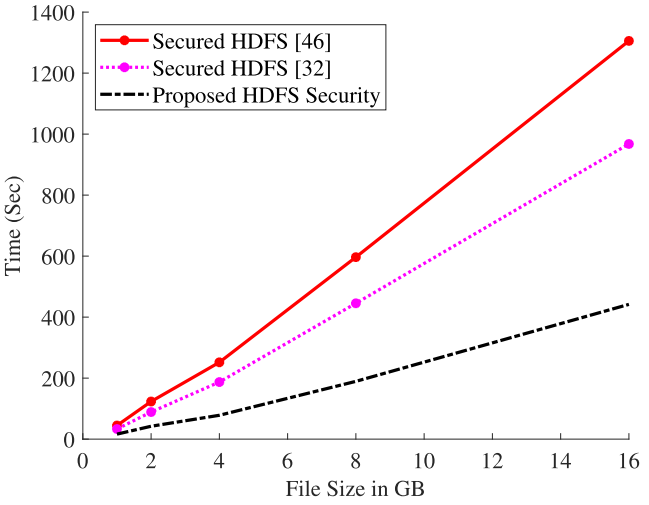

FIGURE 4. Secured data transmission delay time.

From the table, we note that the data transmission response time in our proposed method is the closest to the baseline case and hence proved superiority over the methods in comparison. This improves the data mobility performance in secured cloud systems where the extracting and loading times are considered the same among all secured methods. The results of the experiments are shown in Figure 3 that presents the average response times achieved by HDFS security methods.

3) SECURED DATA TRANSMISSION DELAY TIME

We define the Secured Data Transmission Delay Time (*SDTDT*) as the difference between the secured data transmission response time (*SDTRT*) and the non-secured baseline data transmission response time (*BDTRT*), and then *SDTDT* is computed in Equation (4):

$$SDTDT = SDTRT - BDTRT \tag{4}$$

Figure 4 shows the delay time on big data CSV files caused by the secured HDFS methods.

Based on the Secured data transmission delay time results, the delay time in our security method is the smallest among other secured HDFS methods. However, securing data mobility will affect the cloud system performance and cause throughput degradation.

4) SECURED DATA TRANSMISSION THROUGHPUT

To study the side effects of the secured HDFS methods on big data transmission throughput, we defined the Secured Data Transmission Throughput *SDTT* as the amount of data transmission *ADT* represented by file size in bits from the source cloud to the target cloud divided by Secured Data Transmission Response Time *SDTRT*. The *SDTT* is measured in (Mb/s) and computed as in Equation (5):

$$SDTT = ADT / SDTRT \tag{5}$$

Table 4 presents the throughput (Mb/s) of big data CSV files using HDFS transmission methods.

TABLE 4. Secured data transmission throughput (Mb/s).

Method	1 GB	2 GB	4 GB	8 GB	16 GB
HDFS Baseline [4]	65.484	64.100	59.665	57.332	54.994
Secured HDFS [46]	48.273	43.241	40.909	37.671	35.530
Secured HDFS [32]	51.750	47.573	44.504	41.259	39.113
Proposed secured HDFS	57.894	55.017	52.228	49.183	46.394

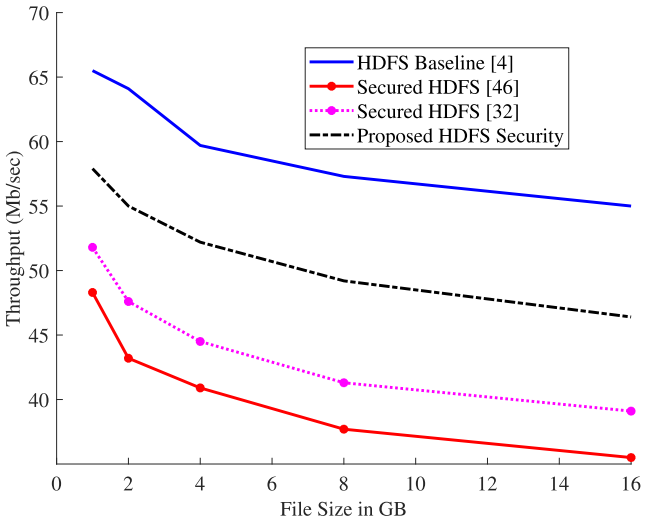

FIGURE 5. Secured data transmission throughput.

Figure 5 presents the throughput of the HDFS security methods

This figure shows that the inter-cloud data transmission throughput decreases as the file size increases. It is clear that all secured HDFS methods cause throughput shortage. However, our method depicts the least throughput shortage among its counterparts. Moreover, this figure implies that the throughput is generally decreased when file size enlarged or doubled. The experiments indicated that HDFS performed best when big data is partitioned into smaller file sizes. It can be inferred from Figures 3, 4, and 5 that our securing algorithm outperforms its counterparts in [32] and [46] for response time, delay time, and throughput.

C. PERFORMANCE EVALUATION

The big data security approach described in [46] proposes a data transmission technique that depend on secure socket layer (SSL) connection between the name nodes of both sender and receiver cloud systems. The cloud receiver name node generates a temporary session key, a random hash number and series of tickets encrypted by the session key to communicate with the cloud sender data nodes, compute the double hash value, and return the encrypted tickets to the sender name node respectively. However, this technique imposes additional security overhead which decreases the cloud system performance mostly when a huge amount of data is transmitted between different clouds.

On the other hand, the security technique introduced in [32] ignores the acknowledgment through the communication process between the sender and receiver cloud data nodes that allows the attacker to drop data packets and may cause serious data threats. If no acknowledgment is assumed after sending the data from the sender to the receiver cloud, the sender data node might delete the data after being transmitted without any knowledge whether it has been correctly received. Nevertheless, this technique enables the attacker to create considerable data lost, drops the packets that the sender data node deletes them, and the receiver did not get any data packet. In addition, retransmission of data packets will increase the network bandwidth as well as the data processing overhead.

Therefore, these secure transmission techniques described in [32] and [46] cause unnecessary network bandwidth and processing overhead due to the extra transmission of data packets and extra encryptions, which increase both the total transmission response time and the total delay time respectively. Consequently, the throughput will be decreased according to equation 5, and hence degrades the HDFS performance of executing data mobility between the clouds.

In contrast, our secure data mobility protocol optimizes the communications, the encryption, and the decryption operations between the sender and the receiver clouds. It requires that critical data are always stored in encrypted format with a key known only to the data owner. Otherwise, integrity check would be sufficient during non-critical data transmission when dealing with public clouds. This requirement greatly enhances the data mobility efficiency, and decreases both the total transmission response time and the total delay time as the sender and receiver engines would not be responsible for encrypting and decrypting large chunks of data while being analyzed, duplicated, or transmitted.

The sender data node decrypts the block access tokens and sends the user's data along with hash value of the data, concatenated with a random nonce to prevent replay attacks. The sender data node only encrypts the hash value because user's data is already stored in encrypted form. Therefore, data owner encryption ensures data confidentiality while our secure data mobility protocol ensures data integrity.

The big data security needs are addressed in our approach that takes into consideration the efficiency and security of data mobility communication processes, the privacy of the users, and the confidentiality of the data. The communication between the user, the sender, and the receiver clouds is protected by the shared credentials between the user and the sender/receiver cloud. The data threat that try to utilize data mobility process to cause data loss is handled by an acknowledgment mechanism which we introduce to ensure that data packets at the sender cloud are only removed after the receiver cloud successfully received it. Moreover, the encryption of the metadata in our protocol also helps to guard against some active attackers. In addition, the change of data packets during transmission or block them in the middle is prevented by verifying the data integrity through the associated hash value and the successful reception of the data that is verified by the returned acknowledgment.

VI. CONCLUSION AND FUTURE WORK

An approach of two fold techniques; big data classification and security, was presented in order to offer data protection controls, avoid threats and risks, and achieving high data mobility in the cloud system. The big data classification technique was introduced based on the criticality of the data. The architecture of the HDFS MapReduce function is used for splitting the file into disjoint fragments and distribute them among different mappers for easy checking of sensitive data. Thus, the proposed classification technique identifies the files that need to be secured and reduces the extra cost when applying the data security to public files which in turn enhances the cloud system performance.

The big data security technique was introduced and described based on big data classification technique. In this technique, only the files that are identified by the classification technique as confidential should be secured before any data transmission over cloud nodes occur

The efficiency that was achieved by the proposed methodology techniques are demonstrated by means of tabular and graphical representations. The technique of big data classification is evaluated in terms of data classification time. The classification method shows high performance improvement by avoiding the redundant encryption and decryption processes for public files. The technique of big data security is evaluated in terms of response time, throughput, and delay time. The evaluation results show the applicability and usefulness of the proposed integrated methodology in protecting confidential files while transmission over different cloud nodes.

A software tool is developed to classify and secure files distributed among several nodes in a cloud system and effectively enhance the system performance in less time. This tool is used to define the metadata of files to facilitate the data security processes. Extensive experimental analysis was conducted over real-world applications in cloud computing systems which indicate that high performance can be achieved by using the proposed classification and security methodology.

In the future, an adaptive method to incorporate security constraints during big data classification will have to be considered in designing and developing cloud computing systems. In addition, we will consider image, video and audio files for classification. However, these types need special treatments as its nature is different from txt, csv, log, xls, and sql files. Therefore, new techniques are needed so that it will be able to deal with file attributes for such big data types.

REFERENCES

[1] M. Paryasto, A. Alamsyah, B. Rahardjo, and M. Kuspriyanto, "Big-data security management issues," in *Proc. 2nd Int. Conf. Inf. Commun. Technol. (ICoICT)*, May 2014, pp. 59–63.
[2] A. K. Tiwari, H. Chaudhary, and S. Yadav, "A review on big data and its security," in *Proc. Int. Conf. Innov. Inf., Embedded Commun. Syst. (ICIIECS)*, 2015, pp. 1–5.
[3] *Sonic*. Accessed: Sep. 2018. [Online]. Available: http://mirrors.sonic.net/apache/hadoop/common/hadoop2.6.0/
[4] K. Shvachko, H. Radia, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Proc. IEEE 26th Symp. Mass Storage Syst. Technol. (MSST)*, May 2010, pp. 1–10.
[5] J. V. Gautam, H. B. Prajapati, V. K. Dabhi, and S. Chaudhary, "A survey on job scheduling algorithms in big data processing," in *Proc. IEEE Int. Conf. Electr., Comput. Commun. Technol. (ICECCT)*, Mar. 2015, pp. 1–11.
[6] A. Holmes, *Hadoop in Practice*. Shelter Island, NY, USA: Manning Publications, 2012.
[7] A. Sinha and P. K. Jana, "A hybrid mapreduce-based k-means clustering using genetic algorithm for distributed datasets," *J. Supercomput.*, vol. 74, no. 4, pp. 1562–1579, 2018.
[8] A. Nasridinov and Y.-H. Park, "Visual analytics for big data using R," in *Proc. 3rd Int. Conf. Cloud Green Comput. (CGC)*, 2013, pp. 564–565.
[9] S.-H. Kim, J.-H. Eom, and T.-M. Chung, "Big data security hardening methodology using attributes relationship," in *Proc. Int. Conf. Inf. Sci. Appl. (ICISA)*, Jun. 2013, pp. 1–2.
[10] C. Tankard, "Big data security," *Netw. Secur.*, vol. 7, no. 7, pp. 5–8, 2012.
[11] N. Chaudhari and S. Srivastava, "Big data security issues and challenges," in *Proc. Int. Conf. Comput., Commun. Autom. (ICCCA)*, 2016, pp. 60–64.
[12] A. Katal, M. Wazid, and R. Goudar, "Big data: Issues, challenges, tools and good practices," in *Proc. 2th Int. Conf. Contemp. Comput. (IC3)*, 2013, pp. 404–409.
[13] S. Sagiroglu and D. Sinanc, "Big data: A review," in *Proc. Int. Conf. Collaboration Technol. Syst. (CTS)*, 2013, pp. 42–47.
[14] T. Mahmood and U. Afzal, "Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools," in *Proc. 2nd Nat. Conf. Inf. Assurance (NCIA)*, 2013, pp. 129–134.
[15] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Netw.*, vol. 28, no. 4, pp. 46–50, Jul./Aug. 2014.
[16] A. Khalid and M. Shahbaz, "Adaptive deadline-aware scheme (ADAS) for data migration between cloud and fog layers," *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 3, pp. 1002–1015, 2018.
[17] T. Payton and T. Claypoole, *Privacy in the Age of Big Data: Recognizing Threats, Defending Your Rights, and Protecting Your Family*. Rowman & Littlefield, 2014.
[18] K. Davis, *Ethics Big Data: Balancing Risk Innovation*. Newton, MA, USA: O'Reilly Media, Inc., 2012.
[19] Y. Gahi, M. Guennoun, and H. T. Mouftah, "Big data analytics: Security and privacy challenges," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2016, pp. 952–957.
[20] R. Alguliyev and Y. Imamverdiyev, "Big data: Big promises for information security," in *Proc. IEEE 8th Int. Conf. Appl. Inf. Commun. Technol. (AICT)*, Oct. 2014, pp. 1–4.
[21] E. Bertino and E. Ferrari, "Big data security and privacy," in *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years*. Cham, Switzerland: Springer, 2018, pp. 425–439.
[22] T. Zaki, M. S. Uddin, M. M. Hasan, and M. N. Islam, "Security threats for big data: A study on Enron e-mail dataset," in *Proc. Int. Conf. Res. Innov. Inf. Syst. (ICRIIS)*, Jul. 2017, pp. 1–6.
[23] S. Marchal, X. Jiang, R. State, and T. Engel, "A big data architecture for large scale security monitoring," in *Proc. IEEE Int. Congr. Big Data*, Jun./Jul. 2014, pp. 56–63.

[24] P. Adluru, S. S. Datla, and X. Zhang, "Hadoop eco system for big data security and privacy," in *Proc. Long Island Syst., Appl. Technol. Conf. (LISAT)*, 2015, pp. 1–6.

[25] G. Raj, R. C. Kesireddi, and S. Gupta, "Enhancement of security mechanism for confidential data using AES-128, 192 and 256bit encryption in cloud," in *Proc. 1st Int. Conf. Next Gener. Comput. Technol. (NGCT)*, Sep. 2015, pp. 374–378.

[26] A. M. Borkar, R. V. Kshirsagar, and M. Vyawahare, "FPGA implementation of AES algorithm," in *Proc. 3rd Int. Conf. Electron. Comput. Technol. (ICECT)*, vol. 3, 2011, pp. 401–405.

[27] V. Gadepally *et al.*, "Computing on masked data to improve the security of big data," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2015, pp. 1–6.

[28] K. S. Arvind and R. Manimegalai, "Secure data classification using superior naive classifier in agent based mobile cloud computing," *Cluster Comput.*, vol. 20, no. 2, pp. 1535–1542, 2017.

[29] E. Damiani, "Toward big data risk analysis," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Oct./Nov. 2015, pp. 1905–1909.

[30] A. Gupta, A. Verma, P. Kalra, and L. Kumar, "Big data: A security compliance model," in *Proc. Conf. IT Bus. Ind. Government (CSIBIG)*, 2014, pp. 1–5.

[31] S.-H. Kim, N.-U. Kim, and T.-M. Chung, "Attribute relationship evaluation methodology for big data security," in *Proc. Int. Conf. IT Converg. Secur. (ICITCS)*, 2013, pp. 1–4.

[32] Q. Shen, L. Zhang, X. Yang, Y. Yang, Z. Wu, and Y. Zhang, "SecDM: Securing data migration between cloud storage systems," in *Proc. IEEE 9th Int. Conf. Dependable, Autonomic Secure Comput. (DASC)*, Dec. 2011, pp. 636–641.

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

[34] D. Miller, S. Harris, A. Harper, S. VanDyke, and C. Blask, *Security Information and Event Management (SIEM) Implementation*. New York, NY, USA: McGraw-Hill, 2011.

[35] N. I. Readshaw, J. Ramanathan, and G. G. Bray, "Method and apparatus for associating data loss protection (DLP) policies with endpoints," U.S. Patent 9 311 495, Apr. 12, 2016.

[36] A. Al-Shomrani, F. Fathy, and K. Jambi, "Policy enforcement for big data security," in *Proc. 2nd Int. Conf. Anti-Cyber Crimes (ICACC)*, 2017, pp. 70–74.

[37] B. Cruz. *Vulnerability, Exposure, Threat and Risk Terms*. Accessed: Sep. 2018. [Online]. Available: http://belencruz.com/en/2013/04/vulnerability-exposure-threat-and-risk-terms/

[38] T. M. Corporation. *Common Vulnerabilities and Exposures*. Accessed: Aug. 2018. [Online]. Available: https://cve.mitre.org/cve/

[39] L. Hayden, *IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data*. New York, NY, USA: McGraw-Hill, 2010.

[40] A. Leitner and I. Schaumuller-Bichl, "ARiMA—A new approach to implement ISO/IEC 27005," in *Proc. 2nd Int. Logistics Ind. Inform. (LINDI)*, 2009, pp. 1–6.

[41] Wikipedia. *Metadata*. Accessed: Aug. 2018. [Online]. Available: https://en.wikipedia.org/wiki/Metadata

[42] Hadoop. *Mapreduce Tutorial*. Accessed: Sep. 2018. [Online]. Available: https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html/

[43] I. Hababeh. (2015). "Data migration among different clouds." [Online]. Available: https://arxiv.org/abs/1512.08383

[44] S. Alouneh, I. Hababeh, and T. Alajrami, "Toward big data analysis to improve enterprise information security," in *Proc. 10th Int. ACM Conf. Manage. Digit. EcoSyst.*, 2018, pp. 106–109.

[45] S. Tatham. *PuTTY: A Free SSH and Telnet Client*. Accessed: Sep. 2018. [Online]. Available: https://www.chiark.greenend.org.uk/~sgtatham/putty

[46] C. Zhonghan, Z. Diming, H. Hao, and Q. Zhenjiang, "Design and implementation of data encryption in cloud based on HDFS," in *Proc. Int. Workshop Cloud Comput. Inf. Secur. (CCIS)*, 2013, pp. 274–277.

ISMAIL HABABEH received the B.Sc. degree from the University of Jordan, Jordan, the M.S. degree from Western Michigan University, USA, and the Ph.D. degree in computer science from Leeds Beckett University, U.K., all in computer science. He is currently an Associate Professor with the Computer Science Department, German Jordanian University. His research interests include cloud computing, big data security, wireless networks, and systems performance.

AMMAR GHARAIBEH received the B.S. degree (Hons.) from the Jordan University of Science and Technology in 2006, the M.S. degree in computer engineering from Texas A&M University in 2009, and the Ph.D. degree in computer engineering from the New Jersey Institute of Technology. He is currently an Assistant Professor with the Computer Engineering Department, German Jordanian University. His research interests include wireless networks and network analysis and design.

SAMER NOFAL received the B.Sc. degree from Yarmouk University, Jordan, the M.S. degree from Hashemite University, Jordan, and the Ph.D. degree from the University of Liverpool, U.K. He is currently an Assistant Professor with the Computer Science Department, German Jordanian University. His research interests include graph algorithms and their applications within intelligent systems research.

ISSA KHALIL received the Ph.D. degree in computer engineering from Purdue University, USA, in 2007. He is currently a Principal Scientist with the Cyber Security Group, Qatar Computing Research Institute, and also a member of Qatar Foundation. His research interests include wireless and wireline network security and privacy. He is especially interested in security data analytics, network security, AI security, and private data sharing. His novel techniques to discover malicious domains following the guilt-by-association social principle attract the attention of local media and stakeholders. His research in detecting and predicting threat indicators through data analytics has been translated into a full-fledged system for threat intelligence generation and sharing that can be consumed by end users and SOC/NOC operators. He is a Senior Member of IEEE and a member of ACM. He received the Best Paper Award in CODASPY 2018. He served as an Organizer, a Technical Program Committee Member, and a Reviewer for many international conferences and journals. He delivers invited talks and keynotes in many local and international forums.