Towards Securing Big Data on Software Defined Network: Performance Aware Architecture Design

Ahmed Mohammed Alghamdi
Department of Software Engineering
College of Computer Science and Engineering
University of Jeddah
Jeddah, Saudi Arabia
E-mail: amalghamdi@uj.edu.sa

Abstract—**Big data security and privacy have been the main concern because many organizations have started to depend on big data for their operations. Big data refers to a large amount of structure and unstructured data that has been defined differently, but, in general, this term refers to the collection of data sets, which has special characteristics. Big data security faces the need to protect sensitive data by using different technologies and policies. It has been argued that there is no comprehensive security solution for big data, however, to achieve this comprehensive solution, parts of big data should be protected and secured. In addition, Software Defined Network (SDN) has gained more interest due to its advantages in improving network management, as well as monitoring with more programmability and better network resource utilization. Many researches have been done in this field, but the trade-off between security and performance has not been considered, especially considering SDN. In this paper, an architecture design is proposed for big data security, which considers security and performance trade-off. The proposed architecture is based on giving each part of big data the proper security mechanism to protect them efficiently, which not only improves the performance, but also saves resources.**

Keywords-Big Data; Big data security; Hadoop; Software Defined Network.

I. INTRODUCTION

In recent years, big data has become increasingly one of the hot topics in Information Technology (IT) research society. This importance comes from the various data usages as well as its analysis and huge size. According to Big Data statistics, data has increased 300 times to be 40,000 Exabytes in 2020 and the Big Data market is currently worth $138.9 billion [1]. The uses of big data and its analysis have attracted information science researchers, decision-makers in public and private sectors, healthcare systems, and IT companies. In addition, Software Defined Networking (SDN) has been recently gaining more interest due to many features that are offered by SDN, which improves the network management and resource utilization. According to Statistics MRC, the "World Software Defined Networking (SDN) Market accounted for $10.88 billion in 2015 and is projected to rise to $134.51 billion by 2022 at a CAGR of 43.2% from 2015 to 2022", which is very high [2].

SDN provides many advantages including programmable network access, large and complex data traffic management, reduced network hardware capital and operating costs, and personalized data control, which has inspired companies to embrace this technology. SDN is a layered network architecture offering unparalleled programmability, automation, and network control by the ramification of the network's control plane and data plane [2]. The network knowledge and states are logically centralized in the SDN architecture, and the underlying network infrastructure for network applications is abstracted. One of the key advantages of this approach is that it offers a more organized software framework for the creation of network-wide abstractions while simplifying the data plane capacity.

Big data refers to any large amount of structured and unstructured data. There are various explanations of big data via Vs, which range from 3 to 6 Vs. Typically, 5 Vs used to characterize the big data including; volume, velocity, variety, veracity, and value. The volume is the data size; velocity is the speed of generating and changing the data; variety is the data in many forms; veracity is accuracy and validity of the data; and value, which provides output from large data set [1][3][4]. Big data have many challenges and difficulties due to their characteristics. It is also important to emphasize that big data can be used for critical decision-making and sensitive tasks; as a result, data trustworthiness is a critical requirement [5]. Data must be protected from unauthorized access and modifications, accurate, complete, and up-to-date. Big data security can be seen from three main aspects including; confidentiality, integrity, and availability. Many components of big data need to be secure including; the data itself during storage, transferring, processing, and the value extracted from these data. Also, other hardware and software components as well as the cloud providers and big data platforms. However, a comprehensive security solution is difficult to achieve in big data.

The rest of this paper is structured as follows. Section 2 discusses big data definitions, characteristics, as well as SDN related aspects. In Section 3, a literature review of big data security challenges and solutions will be shown, as well as SDN-big data related researches. In Section 4, the proposed architecture will be displayed and described. Section 5, evaluation and comparative study are discussed and

 ISBN: 978-1-61208-786-3

compared. Finally, in Section 6, we present the conclusion as well as the future works.

II. BACKGROUND

In this section, two main aspects of our research will be presented to give an overview of them including big data and SDN as the following:

A. Big Data

Although the term "Big Data" has become increasingly common, its meaning is not always clear. Big data has many definitions and explanations, but in general, this term refers to the collection of data sets which has special characteristics including big volume and variety, as a result, it is difficult to deal with such data by using traditional tools of data management and processing. According to Gartner IT glossary [6], the term big data is defined as: High-volume, high-velocity and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation. Similarly, Tec America Foundation [7] defines big data as follows: Big data is a term that describes large volumes of high velocity, complex and variable data that require advanced techniques and technologies to enable the capture, storage, distribution, management, and analysis of the information. In terms of the big data characteristics, it often characterizes by three factors: volume, velocity, and variety refer to them as 3 Vs. However, many researches claimed that big data could be characterized by many Vs, usually ranging from 3 to 6 Vs. In this paper, the 5 Vs will be used to characterize big data, as shown in Figure 1, include the following:

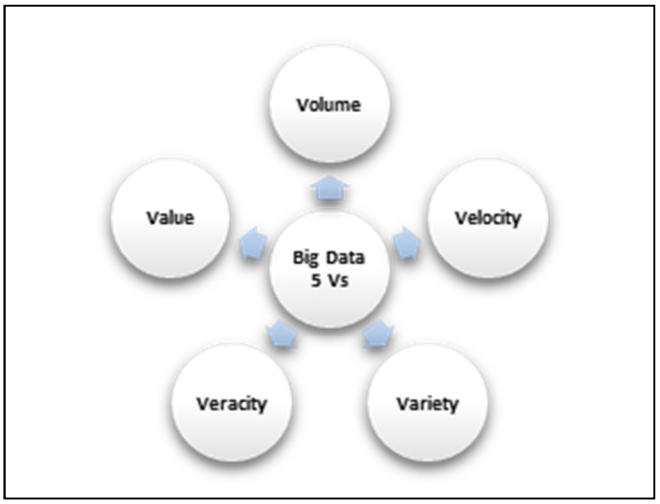

Figure 1. Big Data 5 V's.

1. **Volume**: It refers to the data collected and stored in many distributed systems. It is usually a huge amount of data, which could exceed Exabyte, which can be processed to extract valuable knowledge. The more the data volume increases, the more difficulties for processing with considering performance.
2. **Value**: It is the most important feature of big data that extracting the data value from big data within a specific amount of time. Sometimes the extracted value is more important than the data itself and it has meaning and uses more than using the data before processing.
3. **Veracity**: The validity and accuracy of the collected data have major importance. The quality of Big Data may be good, bad, or undefined due to data inconsistency, incompleteness, ambiguities, and latency. As a result, extracting knowledge or values cannot be occurred from invalid or inaccurate data or might lead to false interpretation. Because of that, collected data need to be checked and any doubt about gathering data should be removed.
4. **Variety**: It refers to the variety of the data types; data could be structured, unstructured, and semi-structured. It also could be internal or external; the internal data is gathered from internal resources in the organization, whereas the external data is gathered from sources. This variety allows processors to extract as interesting as varied information about a specific topic.
5. **Velocity**: It refers to how fast data is being produced and changed and the speed with which data must be received, understood, and processed. Big data does not only rely on static record but it also uses real-time streams and without storage. Processing big data has to be able to generate and extract the results in a few seconds or few milliseconds. Even a few seconds is too late for some critical applications.

Despite the importance of big data, this can lead to a revaluation in organizations and enterprises. The previous five criteria brought the needs to find tools and mechanisms for efficiently processing and analyzing the data. In terms of the big data systems, Hadoop is one of the most popular big data systems, which used to store and process big data. Hadoop is an open-source software used for big data, because of its ability to deal with a very big amount of distributed data. According to The Apache Software Foundation [8], Hadoop is defined as: A framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. Hadoop has many modules including the following [8]:

- **Hadoop Common**: The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS):** A distributed file system that provides high-throughput access to application data. Hadoop uses a block-structured distributed file system for storing a large amount of data called the Hadoop Distributed File System (HDFS). All the individual files in HDFS are divided into blocks with fixed sizes. A cluster of machines with storage capacity is used to store these blocks. The major components of HDFS are NameNode, DataNode, and BackUpNode.

 ISBN: 978-1-61208-786-3

- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

Big data has many security issues and challenges as well as privacy concerns, which must be taken into consideration before building a big data environment. The following are some of the most important challenges that should be considered when dealing with big data:

- Access Control
- Communication Security
- Data Integrity
- Computations Security
- Privacy
- Random Distribution
- Cloud Security
- Hadoop Security
- End-Point input validation and filtering

Traditional security solutions are insufficient when dealing with big data to ensure security and privacy. Encryption techniques, access permissions, firewalls, transport layer security can be broken. For these reasons, advanced techniques and technologies are needed to protect, monitor, and audit big data in terms of data, applications, and infrastructures.

B. Software Defined Network (SDN)

SDN has been defined as an emerging network architecture designed to improve and simplify network management as well as improve network resource utilization. SDN can be viewed in three different layers including data plane, control plane, and application plane. This separation of network devices from their management enables the network control to be programmable, independently developed, and have a flexible design compared to the traditional network architectures [2]. The following Figure 2 shows the SDN architecture.

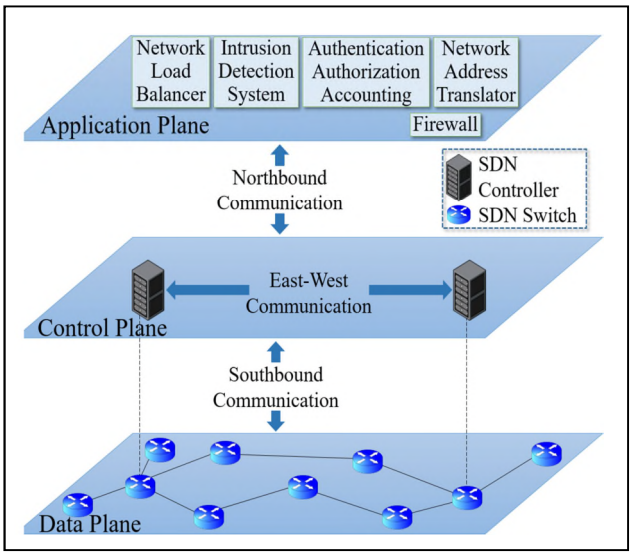

Figure 2. SDN Architecture [9].

The first layer of the SDN architecture is a data plane that includes switches, whether physical and virtual switches, that are considered as forwarding devices. The switches give a view of the programmable flow tables that can describe an operation related to a particular flow for each packet. The second layer of the SDN architecture is the control plane that moves the control logic to an external body, called the SDN controller, which lies within the architecture's control plane. The controller is a software interface that has a full view of the network and the ability to make optimal routing decisions, thereby increasing the visibility of the network. The network is programmable by application software programs located in the third layer, named application plane, that runs at the top of the control plane. This plane has a series of applications that enforce certain functions of network control, such as routing, load balancing, fault tolerance, recovery, etc. By means of a well-defined programming interface between the switches and the SDN controller, the separation of the control plane and the data plane can be understood. The controller controls the elements of the data plane directly through a well-defined Application Programming Interface (API), as shown in Figure 2, the so-called Southbound API.

SDN provides various features for all SDN-enabled devices, such as centralized and decentralized control of multiple cross-vendor network components, primarily data plane platforms with a specific abstraction layer of APIs. It decreases the difficulty of network configuration and operation achieved through the automation of high-level network feature configuration and forwarding behavior [7]. SDN enables fast implementation of new protocols and network-services leading to the high abstraction of operations. The SDN infrastructure can be tailored to the specific user application running on it through a control plane, which improves the user experience considerably.

SDN, however, has its disadvantages: the added flexibility and functionality allow additional overhead on the equipment and, as a result, processing speed and throughput capacity are forfeited. It does not mean that the overall efficiency is automatically decreasing; the SDN-enabled equipment will perform many network services and tasks performed by the end-nodes of the control layers of the network systems in a simpler and faster manner.

III. RELATED WORK

Several researches have been done in big data security and privacy, yet there is no comprehensive security architecture for big data. Because it is impossible to protect all big data and its attributes, big data security has been seen from a different perspective. Many researches and techniques have been deployed and implemented to improve big data security. Some related works are presented as the following:

In the research published in [10], an access control schemes for Hadoop data storage has been proposed based on concepts from BitTorrent and the secure sharing storage over the cloud. Their paper has described the Hadoop architecture as well as the process flows. The security risks that faced Hadoop have been reviewed and a solution for securing data stored on Hadoop over cloud systems has been

 ISBN: 978-1-61208-786-3

proposed. This proposed solution is based on creating and distributing access token over a web server by encrypting the meta-data records from the Hadoop client and creating access token file and encrypting them as well. Finally, the encrypted access token files are distributed to the cloud storage providers. This solution has not been implemented or evaluated in terms of performance and applications for access control.

A dynamic adaptive access control scheme for Hadoop platform has been proposed in [11] that platform that adopts user suspicious status evaluation and user authorization policy based on labels and attributes. This scheme can realize the real-time dynamic adjustment of user authority according to user behavior by designing the trigger mechanism of authority automatic change, thereby more effectively protecting user sensitive information and private data in a big data environment.

A new architecture for securing MapReduce computation in the cloud has been introduced in [12] aiming to secure the Tag-MapReduce framework providing high secure MapReduce computation in the cloud with low overhead. In this paper data integrity, verification, and privacy have been focused on. Security challenges have been discussed and presented for big data processing using MapReduce. Their architecture based on a hybrid cloud and the MapReduce will move to the cloud especially the public cloud which makes it insecure. Their design not only secures the MapReduce but also considering the overhead as well as avoiding some vulnerability. A comparison between the architecture with the previous solutions has been done.

The paper published in [13] has presented a meta-model for security policies and a comprehensive framework for access management at the Infrastructure as a Service (IaaS) level. The proposed framework is being implemented on the open-source IaaS platform OpenStack, using HDFS and MySQL for data storage and adopting IBE as the encryption method. This architecture could be divided into two parts: the trusted authority domain and the data center domain. The trusted authority domain includes two components: an identity & key management engine and a policy engine. The data center domain stores the encrypted data. This paper did not include any evaluation or comparative study.

An approach to provide security to unstructured Big Data has been discussed in [14], that developed to give adequate security to the unstructured data by considering the types of the data and their sensitivity levels. They also have shown that data classification concerning sensitivity levels enhances the performance of the system.

The paper published in [15] focuses on the issue of reaching sensitive data by the cloud operators and proposes a novel approach that can efficiently split the file and separately store the data in the distributed cloud servers, in which the data cannot be directly reached by cloud service operators. The proposed scheme is entitled as Security-Aware Efficient Distributed Storage (SAEDS) model, which is mainly supported by the proposed algorithms, named Secure Efficient Data Distributions (SED2) Algorithm and Efficient Data Conflation (EDCon) Algorithm. The main problem solved by their proposed scheme is preventing cloud providers from directly reaching users' original data.

In addition, the SDN security related to big data that in the paper published in [16] security challenges in the SDN network have been addressed. They propose an approach to predict attacks in the SDN networks by applying machine learning techniques instead of using the traditional technique with threshold values, which tend to be problematic due to dynamic environments of SDN. Their proposed method not only predicts the presence of attack but also attack type by a predictive model, which represents the behaviors of the network, particularly under ARP attack, LLDP Attack, or no attack.

The research in [17] proposed a big data analysis-based secure cluster management architecture for the optimized control plane. A security authentication scheme has been also proposed for cluster management to ensure the legality of the data sources. Moreover, ant colony optimization was used to enable a big data analysis scheme and an implementation system was proposed to optimize the control plane. This work is significant in improving the performance and efficiency of applications running in SDN.

An approach to efficient network design and characterization using SDN and Hadoop has been proposed in [18], which shows a method to control characteristics and provide security to the network, which is helpful in capacity planning and attack detection and prevention and several ways.

IV. THE PROPOSED ARCHITECTURE

From the literature review, it is not clear how to have a software architecture design to present a solution for big data security. Our proposed architecture aims to secure the big data, taking into consideration the performance of writing to or reading from big data storing systems. The main two functions of this architecture are writing and reading. Many security aspects have been considered in this architecture including; file policies, fragmentation, and encryption files by using different techniques. In terms of the performance aspect, this architecture divided the load into a number of agents that will improve the performance. In addition, the data classification will improve the performance by giving each file the relative storing process with its respective importance. Finally, this architecture has flexibility because it can deal with any big data storing system. The following Figure 3 shows the proposed architecture, including the following components:

A. Interface Agent

This agent is responsible for displaying the Graphical User Interface (GUI) on the screen and receiving the user name and password from the user as well as displaying the file after retrieving them.

B. Authentication

This agent is responsible for authenticating the user and make sure only the legitimate user can enter the system. This agent checks the user name and password and uses a digital signature to confirm the identity of the user. In addition, a

one-time authentication code can be used for more protection. This agent can also utilize existing authentication techniques, such as OAuth, OTP, etc.

C. Authorization

In this agent, the access rights to files will be controlled. Access control in particular is the main function of this agent. This agent displays the available files, which the user has permissions to access, read, write, etc. It is connected to policy storage that saves all file policies and user permissions. To secure this storage, the authorization agent is connected to the encryption/decryption agent to encrypt the policy storage by using public/private key cryptography. RSA algorithm will be used for this purpose. However, due to heavy computation regarding the encryption, data at rest will be encrypted and once it is not in used. Data that is being processed will only be encrypted once the related processes are terminated. In terms of giving permissions, the user can update or add permissions to his files by using this agent. However, in some cases, there is some concern regarding the access to unwanted parts of the dataset such as Personal Identifiable Information (PII) through utilizing process' global permission. In this case, all events must be logged to a log server for auditing and monitoring any incident that happened intentionally or unintentionally.

D. Main Agent

The main agent is considered as the main menu, which gives the user the ability to choose to write, update, or read files that he has the authority to access. This agent is connected to the write/update agent and read agent as well as the authorization agent.

E. Write/Update Agent

This agent is responsible for writing a new file to the system or updating existing files. It is connected to the meta-data agent to extract information about the exciting files that will be updated or to store meta-data after writing new files.

F. Read Agent

This agent is used to read files from the storing system. This agent is connected to the meta-data, which gives this agent the ability to reach the files and the needed operations if necessary. This agent also connected to the fragmentation/merge agent, which will be used in case of retrieving fragmented files. It will receive merge files from the fragmentation/merge agent and send it to the collector agent. Furthermore, it is connected to the encryption/decryption agent, which allows the read agent to retrieve encrypted files and decrypt them by using this agent and send them to the collector agent.

G. Data Classification Agent

This agent will receive the user's choice of his data and tag the files with the security classification code, which differentiate the files by its criticality. In this architecture, the data has been classified into:

1. **Sensitive Data**: The data is valuable and need the highest level of security with a different type of protection techniques. The strongest algorithms and standards will be used in this class of data to provide protection. These data might be related to national security, military secrets, or industrial secrets.
2. **Confidential Data**: this class of data has a middle level of sensitivity, needs security algorithms with good processing speed, and might use algorithms less strong than the algorithms used in the sensitive data. This class of data may include data related to military equipment, country political and economic situations, or new changes in the company's future.
3. **Public Data**: This class of data will be open for everyone or give access for registered users using id and password, such as files on the university websites given to the university students.

By using this classification, an adequate level of security will be provided and enhance the processing performance of the system.

H. Fragmentation/Merge Agent

This agent is responsible for fragmenting files into a random number of fragments with different sizes, for more protection, in terms of the writing process. In terms of the reading process, this agent is responsible for merging files and assembles the file to be complete again. This agent will be used in case of writing or updating sensitive data files. In addition, this agent is connected to the meta-data agent to add or update these files records and keep track of the files' fragments places. Also, this agent is connected to the encryption/decryption agent for encrypting these fragments.

I. Encryption/Decryption Agent

This agent will be used to encrypt confidential files as well as encrypt sensitive data that comes from the fragmentation agent. This agent is also connected to the meta-data and authorization agents to protect their storage and provide security to them. In contrast, the decryption function will be used to decrypt files and send them to the read agent as well as decrypting fragments. Finally, after encrypting the data files, the files will be sent to the storing systems.

J. Meta-Data Agent

This agent is responsible for storing all meta-data of the saved files as well as the new files. This agent is connected to five agents including; write, read, data classification, fragmentation, and encryption agents. It has meta-data storage that has been protected by using encryption algorithms.

K. Collector Agent

This agent will collect files from the storing systems and apply the reading-related functions to them and at the end sending the complete file to the interface agent.

V. Evaluation and Comparative Study

In this section, the evaluation of the proposed architecture will be discussed and compared with some related literature review. In many researches, security issues

and difficulties have been discussed and many solutions have been proposed. However, a comprehensive solution for securing big data is not available yet and considered an important challenge. In addition, some researches focus on securing the big data framework, such as Hadoop, without considering a high-level abstract solution. Furthermore, the trade-off between big data security and performance has not been considered in many researches.

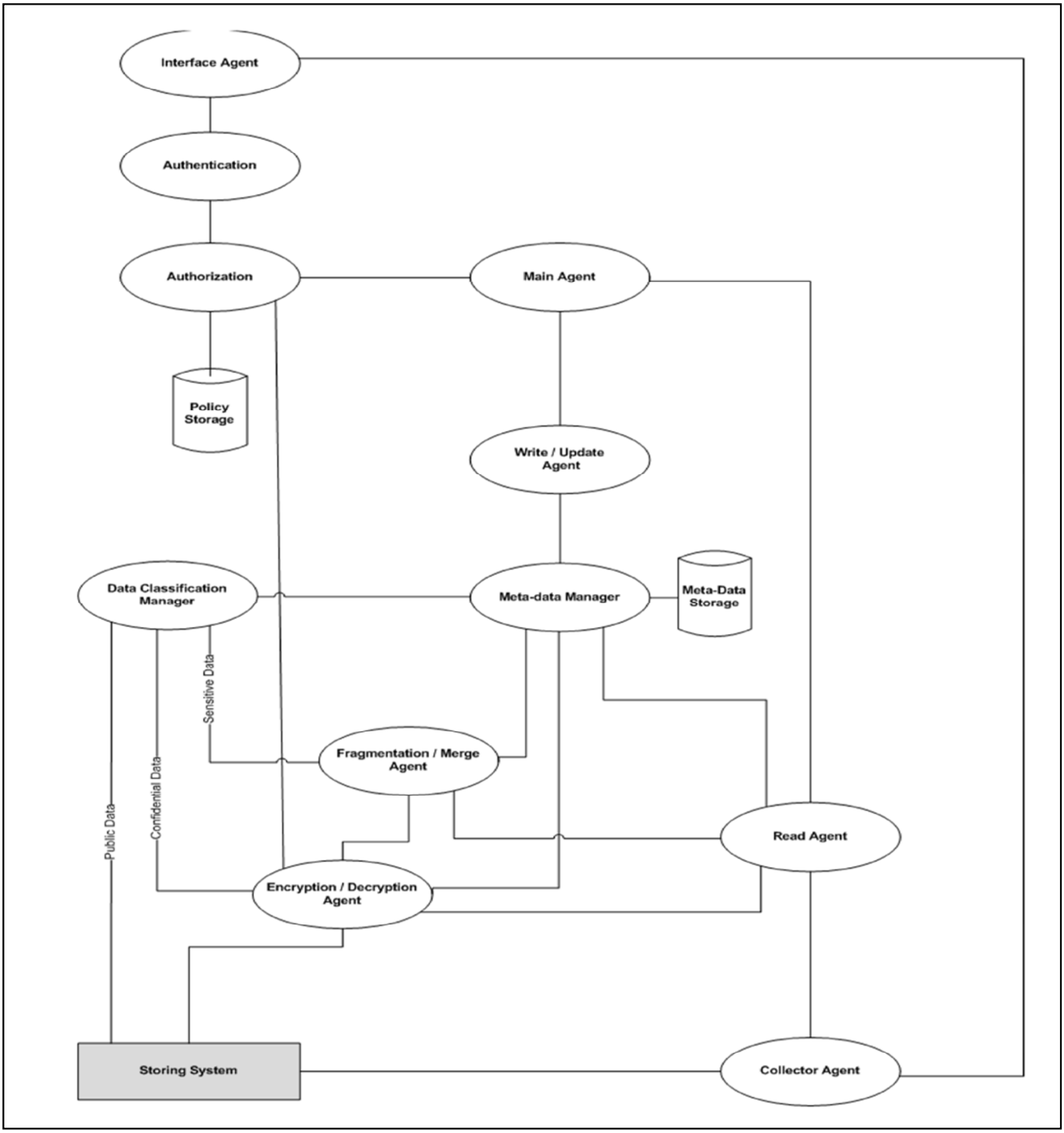

Figure 3. The Proposed Architecture

 ISBN: 978-1-61208-786-3

In the proposed architecture, a high-level solution has been considered, which gives this architecture the ability to work with any big data framework. This could allow this architecture to work with multiple storing systems at the same time without affecting the architecture design or functionality. In terms of performance, this architecture considered the trade-off between security and performance by classifying the data into categories and gives the proper protection to each category. This technique not only improves the performance but it saves the related resources. In addition, the load has been distributed into several agents to improve the performance, such as having the collector agent to take some load from the read agent.

The agent architecture style has been used to benefit from its advantages as well as its mobility. Dealing with a big amount of data, which is a big data characteristic, needs a mobile agent to travel rather than bring all data to the user. An agent architecture is a dynamic architecture that will be created during the runtime and has the ability to run on different software and hardware, which will be needed in a big data environment.

VI. CONCLUSION AND FUTURE WORKS

In this paper, an overview of big data and its related characteristics and security challenges have been presented. Some related works were revised and discussed. An architecture design has been proposed for big data security, which considers security and performance trade-off. The proposed architecture is based on giving each part of big data the proper security mechanism to protect it in a more efficient way. This not only improves the performance, but also saves resources and gives every part what is really needed. In addition, some sequence diagrams have been presented to explain some processes of the proposed architecture. Finally, a comparative study has been done to evaluate the proposed architecture.

For future works, the proposed architecture needs some related works to have a comprehensive security solution for big data. In addition, some performance measurements are needed to discover the architecture performance in different scenarios and to improve any notices regarding the load distributions or the processing time, especially the encryption/decryption processes and fragment/merge processes. Finally, an implementation of this architecture will be needed as well as including some transportation security solutions to improve security.

REFERENCES

[1] D. S. Terzi, R. Terzi, and S. Sagiroglu, “A survey on security and privacy issues in big data,” in 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), 2015, pp. 202–207, doi: 10.1109/ICITST.2015.7412089.

[2] M. Alsaeedi, M. M. Mohamad, and A. A. Al-Roubaiey, “Toward Adaptive and Scalable OpenFlow-SDN Flow Control: A Survey,” IEEE Access, vol. 7, pp. 107346–107379, 2019, doi: 10.1109/ACCESS.2019.2932422.

[3] P. Adluru, S. S. Datla, and X. Zhang, “Hadoop eco system for big data security and privacy,” in 2015 Long Island Systems, Applications and Technology, 2015, pp. 1–6, doi: 10.1109/LISAT.2015.7160211.

[4] Y. Gahi, M. Guennoun, and H. T. Mouftah, “Big Data Analytics: Security and privacy challenges,” in 2016 IEEE Symposium on Computers and Communication (ISCC), 2016, pp. 952–957, doi: 10.1109/ISCC.2016.7543859.

[5] E. Bertino, “Big Data - Security and Privacy,” in 2015 IEEE International Congress on Big Data, 2015, pp. 757–761, doi: 10.1109/BigDataCongress.2015.126.

[6] I. Gartner, “Gartner IT Glossary,” 2016. [Online]. Available: http://www.gartner.com/it-glossary/big-data/. [Accessed: 01-Jul-2020].

[7] S. Mills, S. Lucas, L. Irakliotis, M. Rappa, T. Carlson, and B. Perlowitz, “Demystifying Big Data: A Practical Guide to Transforming the Business of Government,” 2012.

[8] The Apache Software Foundation, “What Is Apache Hadoop?,” The Apache Software Foundation, 2016. [Online]. Available: http://hadoop.apache.org/index.html. [Accessed: 01-Jul-2020].

[9] T. Das, V. Sridharan, and M. Gurusamy, “A Survey on Controller Placement in SDN,” IEEE Commun. Surv. Tutorials, vol. 22, no. 1, pp. 472–503, 2020, doi: 10.1109/COMST.2019.2935453.

[10] C. Rong, Z. Quan, and A. Chakravorty, “On Access Control Schemes for Hadoop Data Storage,” in 2013 International Conference on Cloud Computing and Big Data, 2013, pp. 641–645, doi: 10.1109/CLOUDCOM-ASIA.2013.82.

[11] J. Li, G. Zhao, X. Sun, and Y. Liu, “A Dynamic Adaptive Access Control Scheme for Hadoop Platform,” in 2019 IEEE 2nd International Conference on Computer and Communication Engineering Technology (CCET), 2019, pp. 79–83, doi: 10.1109/CCET48361.2019.8989081.

[12] C. A. A. Bissiriou and M. Zbakh, “Towards Secure Tag-MapReduce Framework in Cloud,” in 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016, pp. 96–104, doi: 10.1109/BigDataSecurity-HPSC-IDS.2016.78.

[13] S. Li, T. Zhang, J. Gao, and Y. Park, “A Sticky Policy Framework for Big Data Security,” in 2015 IEEE First International Conference on Big Data Computing Service and Applications, 2015, pp. 130–137, doi: 10.1109/BigDataService.2015.71.

[14] M. R. Islam and M. E. Islam, “An approach to provide security to unstructured Big Data,” in The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014), 2014, pp. 1–5, doi: 10.1109/SKIMA.2014.7083392.

[15] K. Gai, M. Qiu, and H. Zhao, “Security-Aware Efficient Mass Distributed Storage Approach for Cloud Systems in Big Data,” in 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016, pp. 140–145, doi: 10.1109/BigDataSecurity-HPSC-IDS.2016.68.

[16] E. Unal, S. Sen-Baidya, and R. Hewett, “Towards Prediction of Security Attacks on Software Defined Networks: A Big Data Analytic Approach,” in 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 4582–4588, doi: 10.1109/BigData.2018.8622524.

[17] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, “Big Data Analysis-Based Secure Cluster Management for Optimized Control Plane in Software-Defined Networks,” IEEE Trans. Netw. Serv. Manag., vol. 15, no. 1, pp. 27–38, Mar. 2018, doi: 10.1109/TNSM.2018.2799000.

[18] A. Desai, K. S. Nagegowda, and T. Ninikrishna, “Characterization Using SDN and Hadoop,” 2016 Int. Conf. Circuit, Power Comput. Technol. (ICCPCT), pp. 1–6, 2016, doi: 10.1109/ICCPCT.2016.7530122.

 ISBN: 978-1-61208-786-3