Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2019, 6(8):79-86

Research Article ISSN: 2394 - 658X

Network authentication protocol to securely access the sensitive and clumsy data by using Hadoop for big data

Kartheek Pamarthi

Kartheek.pamarthi@gmail.com

ABSTRACT

The arrival of big data has brought with it new concerns about data security. Some examples of such problems include data dissemination by third parties, the lack of clarity and accuracy in publicly available databases, and the lack of insurance against data breaches and leaks. Mastering the art of effectively exploiting security and hidden measures is a formidable obstacle when dealing with large amounts of distributed data. On the big data platform, this study proposed an architecture that would allow for the storage and accessibility of data that is secure, informative, and high-speed. To ensure that sensitive and cumbersome data may be accessed in a secure manner, we provide a Kerberos network authentication system. Simultaneously with facilitating secure, high-speed access to data, the framework effectively protects, shares, and preserves the owner's data.

Keywords: Big data, security policy, anonymization, encryption, access control, policy enforcement

INTRODUCTION

Big data refers to an infrastructure that enables the management and examination of data sets that are substantially larger than those handled by traditional data processing methods [1]. This structure is also known as big data. Having the skills to mix data from many sources and formats and extract useful information from data sets. Organisations may now gain a more thorough and detailed knowledge of their business thanks to big data, which has changed the way they keep data. This is a strong benefit. 2 [2] in the exponential growth of online communities, media sharing platforms, and IoT devices has resulted in an unprecedented deluge of data [3,]. The current trend of companies collecting more and more data, both in quantity and quality, is not going away anytime soon.

Furthermore, the majority of this data is completely unstructured, which indicates that traditional systems are unable to perform any analysis on it. There is a willingness among the organisations to extract additional beneficial information from the large volume of data and to diversify the data. Here, "Internet of Things" (IoT) means the ever-expanding web of interconnected physical objects that can collect, process, and share data wirelessly over the Internet; this network can function autonomously and without human intervention. The number of "things" currently connected to the Internet of Things (IoT) is 4.95 billion, according to estimates given by Gartner Inc. Scientists predict the figure will rise to 30 billion by 2020, a 35% increase from 2014 [5]. Data will inevitably be generated by each of these devices.

One such framework that Apache created is the Hadoop framework. It enables the distributed processing of colour data sets across computer clusters through the use of programming models [6]. Its scalable design allows it to run on servers ranging from one to thousands, each of which can be utilised independently. Data mining, machine learning, and predictive analytics are just a few examples of the advanced analytics jobs that initially made use of this expanding network of big data tools [7]. This ecosystem is comprised of an ever-expanding variety of technologies. The big data system feeds its data into the Hadoop architecture. Distributed file system (HDFS) developed by Hadoop and deployed across multiple servers is used to store the data.

These servers serve a variety of purposes, such as the NameNode, which is responsible for storing metadata, and the DataNodes, which are responsible for storing application data. On the other hand, Hadoop's primary advantage lies in the fact that it is an open-source implementation of MapReduce [8]. The Hadoop Core Modules is shown in figure 1.

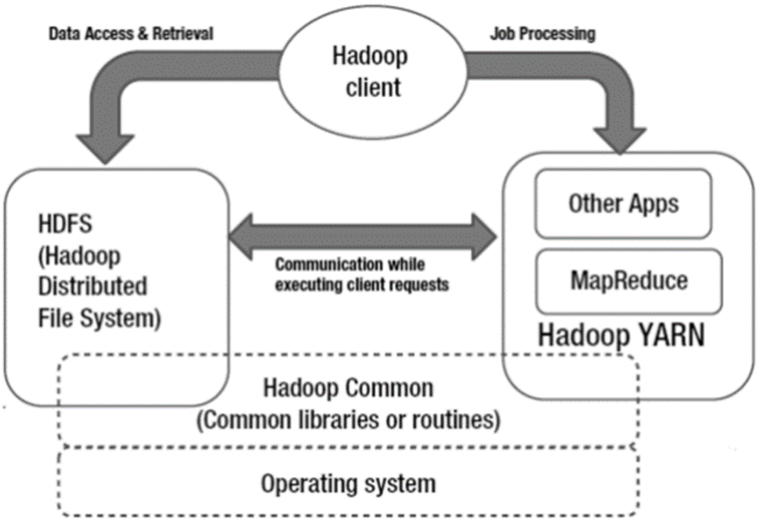
Data Access & Retrieval
Hadoop client
Job Processing
HDFS (Hadoop Distributed File System)
Communication while executing client requests
Other Apps
MapReduce
Hadoop YARN
Hadoop Common (Common libraries or routines)
Operating system

Figure 1. The Hadoop Core Modules.

The MapReduce programming model is a [9] programming paradigm that is primarily geared towards the processing and generation of massive data collections. Traditional security procedures, developed for protecting small-scale, stable data on firewalls and semi-isolated networks, are now insufficient due to the ever-increasing use of streaming cloud technologies for big data [10]. weak tools, dangerous and vulnerable public and private databases, deliberate and accidental data leaks, and weak public and private legislation are only a few of the reasons why hackers can constantly gather resources.

Managing a large data set securely involves a number of issues, some of which are listed here [11]. Even more challenging is the fact that they protect the data while those around them do not trust it. When transitioning from one type of data to another, it is not uncommon for big data technologies to lack extra security and policy certifications [13] and for certain tools to lack such certifications [12]. Problems emerge anytime a new, potentially game-changing technology is released. When discussing big data, these concerns include not just the quantity and variety of data, but also its authenticity and safety [14].

As can be seen in figure 1, the Hadoop common module is responsible for providing shared libraries, and HDFS provides distributed storage as well as the functionality of a fault-tolerant file system capabilities. The functionality of distributed data processing is acquired through the use of YARN or MapReduce. This Hadoop cluster is therefore regarded operational even without all the other features. One of the nodes can be configured to serve as the NameNode, and a few of DataNodes can be added to create a Hadoop cluster that is simple and operating well.

LITERATURE REVIEW

The emergence of Apache Hadoop [14], a system for large-scale data analytics, represents a watershed moment in the history of company growth. To accommodate the ever-increasing demands and prerequisites of business intelligence (BD), the Hadoop community has been hard at work to enhance its stack. Because of its capacity to store and handle a vast quantity of new sorts of data and to make use of modern data architecture, Enterprises across all of the world's main economies have embraced Hadoop. Hadoop keeps a respectable degree of isolation and security while abstracting the management of data, task scheduling, and computing resources.

It is capable of handling a wide range of workloads, including both structured and unstructured workloads.

Incorporating HDFS [15] into a large-scale Hadoop platform is usual practice. This allows the platform to accept commodity hardware and adapt diverse processing frameworks. It accesses the Hadoop environment and manages data via a master/slave architecture. It is a data storage system that several resource schedulers can use, including HTCondor [16] and Spark [17].

A large number of distributed systems also benefit from it. Identity and Access Management (IAM) is a collection of procedures and tools in the HDFS ecosystem that permits end users and applications to safely engage with the system's core capabilities; this, in turn, guarantees accurate data access across the cluster. Authentication, authorization, and identity and access control are the three tiers that make up this security domain. With the increasing number of services and users integrating with Hadoop's federation portfolio, access control has become an increasingly important consideration in the search for a scalable BI hub. The inability to limit access control rules and the enforcement monitor that goes along with them to a standard model is the main challenge when designing an adaptive access control solution for BD platforms. There are currently no established models for integrating application and user access to resources, services, and data in the emerging subject of business intelligence (BD) [18]. By default, Hadoop does not have an integrated classification system that can simplify auditing procedures or enhance information governance. Consequently, there is an increasing amount of written material discussing the difficulties of safely exploiting Hadoop 3.x's fundamental features and the necessity to formalise its Identity and Access Management (IAM) components. The mapping of relevant technologies and the development of new Hadoop capabilities for audit log management and access control frameworks are examples of this trend, which is a kind of knowledge capture.

Hadoop Federation

A Hadoop cluster's NameNode and ResourceManager are subjected to an increasing amount of strain as the cluster's size grows. Both HDFS Federation [19] and YARN Federation [20] have been implemented by Hadoop in order to combat this issue. Each NameNode in HDFS Federation is accountable for the management of a certain portion of the entire namespace. The use of several standalone NameNodes constitutes HDFS Federation. An explanation of the structural design of the single HDFS is important for comprehending the operation of this federated architecture. Hadoop established a universal block storage layer by successfully separating namespaces and blocked storage [21]. Using a federated BD environment helps improve the scalability and isolation of Hadoop processes. This setup stores data in a shared block pool and manages blocks using many independent namespaces. when a result, the authentication need rises when every DN registers with every NN in the cluster. The tightly coupled nature of the namespace and block storage is relaxed to achieve this. When issued, directives from the NNs must be handled by these DNs, and they must also transmit periodic heartbeats and block reports. Because of this, we may aggregate Hadoop clusters located in different parts of the world and also scale the neural network horizontally. This layer is disabled in non-secure Hadoop mode, allowing internal entities to connect directly with the Hadoop services. This includes clients (created by the host operating system), applications, and ecosystems. Conversely, an authentication mechanism can use the Kerberos protocol (token-based authentication) to confirm that an entity is who it says it is with Hadoop secure mode [22].

A KERBEROS*-BASED BIG DATA SECURITY SOLUTION

The Protecting user identities within the Hadoop large data environment is shown in figure 2. Apache Kerby 2.0 includes Intel's Hadoop Authentication Service (HAS), an authentication framework; Alibaba improved the efficiency and accuracy of its Cloud E-MapReduce (EMR) and Cloud HBase user ID information management with its assistance. Administrative costs have also been reduced by the company. A popular secure network authentication protocol, Kerberos, can be easily integrated with pre-existing enterprise identity management systems with the help of the pluggable authentication framework.

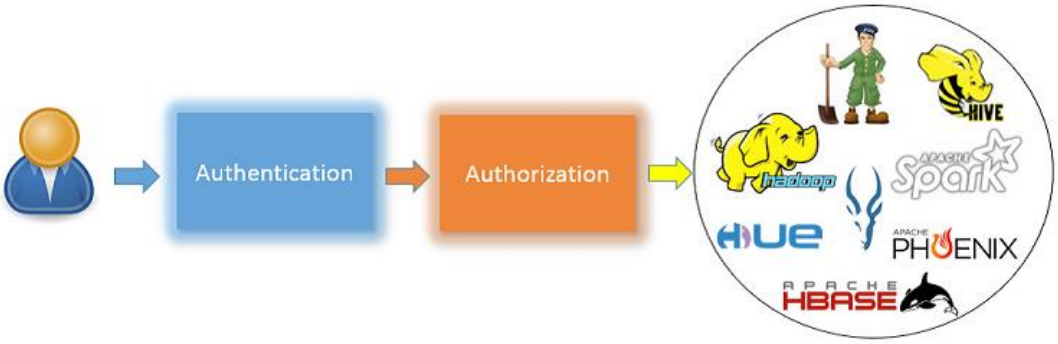

Figure 2. Protecting user identities within the Hadoop large data environment

The only user authentication option that is built-in and safe within the Hadoop big data environment is called Kerberos. The majority of open-source data components include the capability to provide Kerberos authentication for various users and services. Kerberos authentication in big data platforms, on the other hand, presents two issues by itself:

1. Not all potential kinds of encryption and checksums are supported by the Java Runtime Environment (JRE) and the Java Development Kit (JDK). The fact that GSSAPI and SASL are concealed further complicates matters when it comes to additions and changes.
2. Since Hadoop only allows password authentication when using Kerberos, it is tough to integrate an existing identity authentication system with the authentication procedure.

HAS offers a comprehensive authentication solution for the Hadoop open source environment, which allows it to handle the difficulties that have been identified. By connecting with pre-existing authentication and authorization systems, HAS is able to provide additional authentication options in addition to Kerberos on Hadoop/Spark*. The foundation of HAS is a Java application that implements the Kerberos protocol and it was shown in figure 3. Another perk is that it simplifies things and eliminates potential risks since it doesn't require the separate retention of the identifying information.

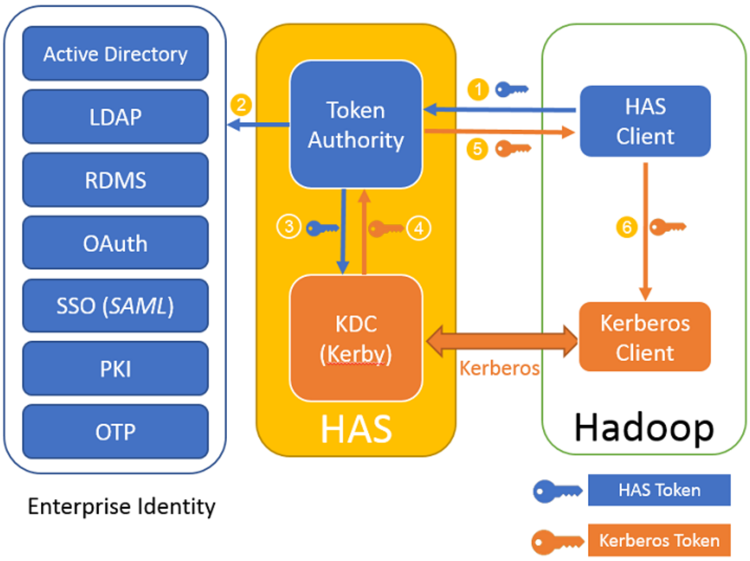

Figure 3. Apache Kerby, a Java implementation of Kerberos, is utilised by HAS to establish a new authentication solution for the Hadoop open source big data environment.*

The main benefits of HAS are:

1. Hadoop services make advantage of the authenticating technique that was first developed by Kerberos. Due to the fact that counterfeit nodes do not get key information in advance, they are unable to communicate with nodes that are already associated with the cluster. The Hadoop cluster is protected from being used in a malicious manner.
2. Users of Hadoop can keep logging in using the same method they've always used. The Kerberos protocol developed by the Massachusetts Institute of Technology (MIT) meets the requirements of HAS compatibility. In addition, users can be authenticated by utilising passwords or Keytab prior to accessing the services that correspond to their credentials.
3. The HAS plugin mechanism establishes a connection with the pre-existing authentication system, and it is possible to construct numerous authentication plugins according to the criteria established by the user.
4. It is not necessary for security administrators to synchronise user account information with the Kerberos database, which results in a reduction in both the expenses of maintenance and the amount of information that is lost.

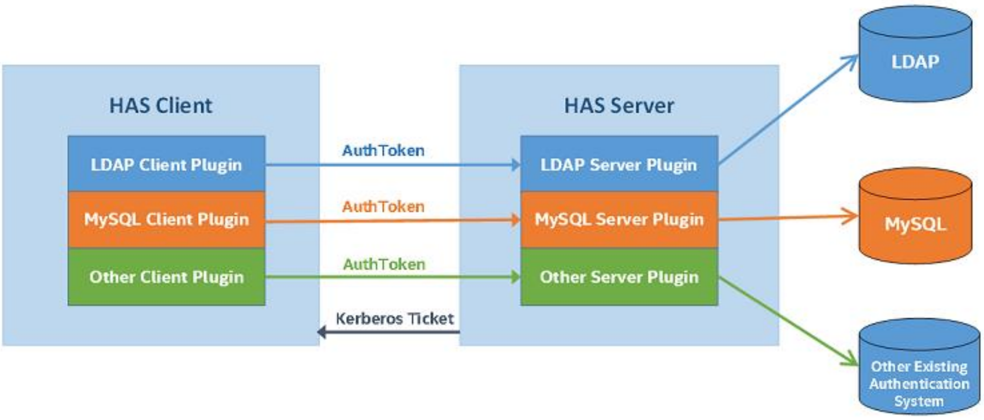

Figure 4: HAS works with LDAP, MySQL, and many other server and client tools.

Because HAS is Kerberos protocol compliant, the Hadoop ecosystem as a whole can utilise the Kerberos authentication method that HAS provides.

The figure 4 HAS offers a number of interfaces and tools that can assist in making deployment more straightforward. It also has user interfaces that can be used to add tools that connect Kerberos to other user identity management systems.

Currently, HAS may be used with LDAP, MySQL, and ZooKeeper. The Apache Kerby group is going to pitch in with the HAS project.

It is anticipated that the HAS feature will be made available in Kerby 2.0, as stated in the community plan.

PROPOSED SECURITY FRAMEWORK MODEL FOR BIG DATA SECURITY

A. Security Framework

We have developed a security architecture to protect Hadoop-based large data collections. More infrastructure compliance and faster data processing are achieved through the utilisation of HDFS, a distributed and parallel processing system. Even though there are various real-time storage solutions available, we chose to use HDFS. Figure 5 shows the framework's architecture. There are two conceptual parts to our design: the enterprise network and the data centre. The engine for the authentication and authorization protocol is now part of the company's network. Using a policy MySQL database and a dynamic one-time code generation engine, the encrypted data is stored in HDFS by the data centre domain. You can store encrypted data in a data centre using databases or HDFS, which is the Hadoop Distributed File System.

Data is encrypted and decrypted using a mechanism to ensure its security. Many other types of data sources are conceivable. When data encryption is first implemented, it is kept in the local file system. Encrypting data using the MapReduce architecture can drastically cut down on the time needed for both encryption and decryption. HDFS is the designated location for storing encrypted data together with the corresponding key. There is a matching security policy in the MySQL policy database that is checked against the encrypted data as well. This makes ensuring the data is safely available while also limiting the user's ability to view it. In addition to managing who has access to the data, the manager of the data centre must also monitor the claims made by interested parties when requesting access.

In the realm of data centres, this is a crucial element.

If the user is validated and accepted, and if the requirement that is enforced by policies is reviewed and completed, then the data will be encrypted and will only be available if and only to the user.

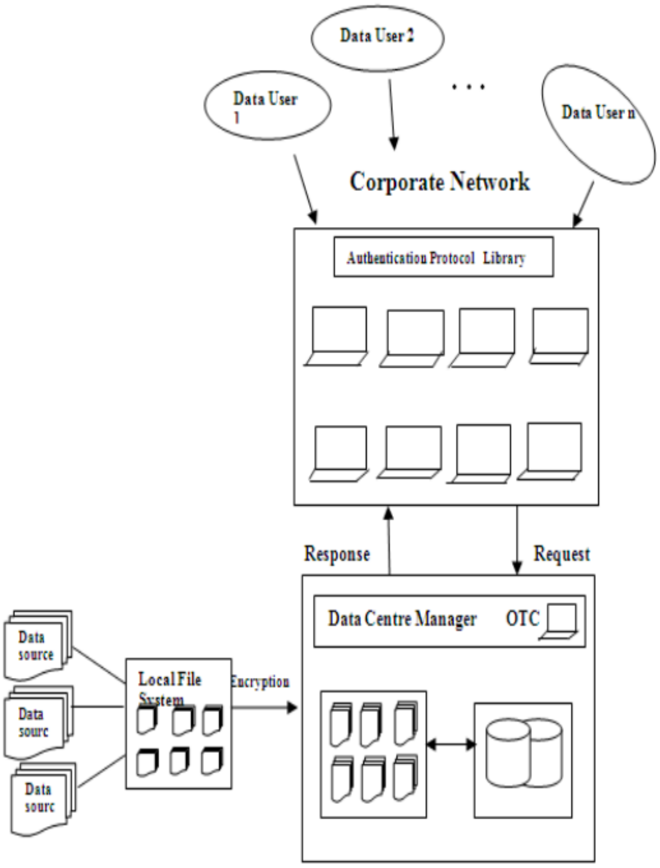

Fig.5. Big data security framework based on Hadoop.

B. Policy Enforcement

A number of cryptographic algorithms can be used to achieve policy enforcement, which is a basic mechanism for controlling rules. Public key encryption (PKE), proxy re-encryption (PRE), type-based pre-encryption (TPRE), and name-based encryption (IBE) are all examples of such approaches.

We are free to use any encryption strategy when it comes to data-regulation mapping. To illustrate, let's look at PKE. The basic idea behind public key encryption (PKE) is that the data centre should control the rules for when the data user gets their private key, the data centre should enforce the rest of the constraints, and the data owner should encrypt data using the data user's public key.

C. Policy Checking

Policy checking is the process of figuring out if a request meets a policy rule. A policy rule's endorsement need can be assessed using this process. The endorsement of a policy regulation usually has the result of "permit" or "deny" in most instances.

When the user request is found to be in compliance with the policy rule, only the user is granted permission to access the data; otherwise, the request is denied. On the other hand, these two effects are not sufficient in particular surroundings and unique conditions. Even though the authority does not have the authorization to carry out the action that is being appealed for, the request is usually acknowledged in order to rescue the patients. For instance, more complicated scenarios can arise in the healthcare industry, particularly in the event of an emergency.

D. Authentication and Authorization

Verifying the identity of a user is known as authentication.

A combination of a user's password and a distinct identifier allows us to verify their identity in most cases. To authenticate, either a human or a system needs to show proof of identification. A user's ability to access certain files and perform certain tasks is not constrained by authentication. The process by which a server or system verifies a user's identity and determines whether they are authorised to access a specific resource or file is known as authorization. A number of alternative authentication protocols are available; a partial list includes Kerberos, RADIUS, DIAMETER, TACACS (Terminal Access Controller Access Control System), and many more.

Because of how user-friendly and straightforward Kerberos is, we have decided to implement it into our Framework. Kerberos is a security protocol that authenticates users on an open network. The client-server architecture is the foundation of this idea. This network authentication protocol uses tickets as its foundation to securely connect nodes or systems over an unsecured network, allowing them to prove their identities to one other.

THE SECURITY CHALLENGES

Among the many capabilities of big data systems are the following: the capacity to store massive amounts of data; the ability to manage and process that data across multiple systems; and the supply of capabilities for data searches, data consistency, and system administration. While large corporations have long been using big data, smaller and medium-sized companies are starting to catch on. This is because big data helps cut costs and makes data handling and management easier. The proliferation of big data has led to the development of a plethora of new IT tools and capabilities.

These tools and skills enable organisations to acquire, handle, manage, and analyse massive volumes of structured and unstructured data in order to gain actionable insights and a competitive advantage. On the other hand, the difficulty of maintaining the confidentiality and safety of sensitive information is brought about by this new technology. Because of the sensitive nature of all of this information and the potential damage that may be caused if it were to fall into the wrong hands, it is imperative that that information be protected from being accessed by unauthorised parties. Furthermore, current security solutions are not capable of effectively managing dynamic data; rather, they are only able to govern static data. In order to achieve this goal, we will be discussing some of the most frequent security concerns in this part. Also the challenges of Big data is shown in figure 6.

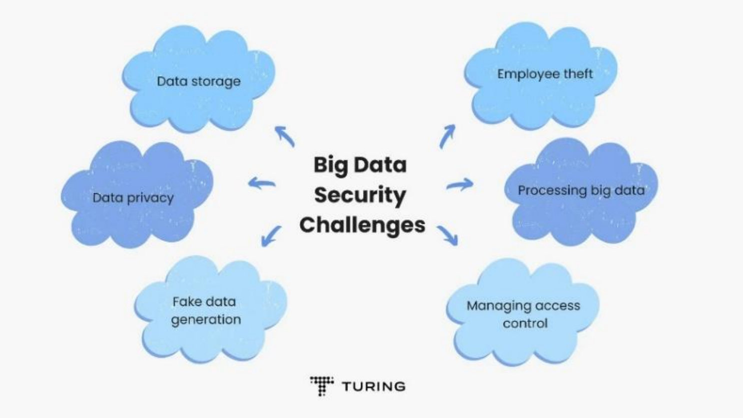

Figure 6: Big data security challenges.

A. Secure Transaction Logs and Data

Different levels of data may be present in a storage media, such as transaction logs and other intelligent information; however, this alone is not sufficient to ensure the integrity of the data. An example of this would be the IT manager gaining visibility into the data that is being moved as a result of the movement of data between these distinct levels. Due to the fact that the quantity of the data is always growing, auto-tiering is a vital component of big data storage management because of its scalability and availability. However, the auto-tiering mechanism does not maintain track of the location of the data storage itself, which presents new issues for the storage of large amounts of data.

B. Monitor, Detect and Resolve Problem

In the absence of the capability to detect non-compliance problems, as well as suspected or actual security breaches, and to rapidly rectify them, even the most ideal security models will be discovered to be lacking. The organisation is responsible for ensuring that the monitoring and detection procedures that are considered to be the best practices are in place.

C. Validation and Filtration of End Point Inputs

When it comes to the maintenance of huge data, the end point devices are the most important aspects. The processing, storage, and other tasks that are required are carried out with the assistance of input data, which is supplied by any end points that are present. Consequently, it is imperative that an organisation makes certain that they are utilising a genuine and legitimate end point gadget.

D. When Generate Information for Big Data

Maintaining a healthy equilibrium between data privacy and data utility is the company's responsibility. Prior to storage, the data should undergo an appropriate level of anonymization, with all user identifiers erased. An inherent security risk may exist due to the fact that unique IDs alone may not be enough to ensure that the data will remain anonymous. Following the de-anonymization procedure, the data that has been anonymized could be cross-referenced with other data that is available.

Real-time compliance and security monitoring

Real-time compliance and security monitoring are of the utmost importance for big data security in this digital era, when businesses are facing the challenge of dealing with an enormous volume of data. By regularly monitoring safety and adherence to compliance concerns, organisations are able to identify potentially harmful behaviours and take the appropriate measures to address them before they affect the organization's operations. Big data is subject to a variety of rules and compliance standards, including the General Data Protection Regulation (GDPR), the Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry Data Security Standard (PCI-DSS). By producing notifications in the event of any infringement, real-time monitoring helps organisations avoid penalties and reputational damage. This is accomplished by ensuring that big data processing activities continue to comply with the requirements.

CONCLUSION

The paper presents a security architecture that considers both the network level (data in motion) and the host level (data at rest) security concerns. Also, using HDFS and MapReduce, we were able to handle the data processing. In addition to meeting the usual authentication requirements, the proposed architecture incorporates extra security measures including data-binding restrictions and a Dynamic One-Time Code. Every possible application can make good use of the suggested framework because of how it was designed. There is room for improvement in the current framework to accommodate more security requirements without major structural modifications. This would allow it to function better with approaches such as attribute-based encryption, the Kerberos authentication protocol, and dynamic one-time codes.

REFERENCES

[1]. Yusuf Perwej, "An Experiential Study of the Big Data," International Transaction of Electrical and Computer Engineers System (ITECES), USA, ISSN (Print): 2373-1273 ISSN (Online): 2373-1281, Vol. 4, No. 1, page 14-25, March 2017, DOI:10.12691/iteces-4-1-3

[2]. V Mayer-Schonberger, K Cukier, Big data: a revolution that will transform how we live work and think, Boston:Houghton Mifflin Harcourt, 2013

[3]. Yusuf Perwej, Mahmoud Ahmed AbouGhaly, Bedine Kerim and Hani Ali Mahmoud Harb,"An Extended Review on Internet of Things (IoT) and its Promising Applications", Communications on Applied Electronics (CAE), ISSN : 2394-4714, Foundation of Computer Science FCS, New York, USA, Volume 9, Number 26, Pages 8– 22, February 2019, DOI: 10.5120/cae2019652812

[4]. Yusuf Perwej, Majzoob K. Omer, Osama E. Sheta, Hani Ali M. Harb, Mohmed S. Adrees, "The Future of Internet of Things (IoT) and Its Empowering Technology" , International Journal of Engineering Science and Computing (IJESC), ISSN: 2321- 3361, Volume 9, Issue No.3, Pages 20192– 20203, March 2019

[5]. Gartner says 4.9 Billion Connected „Things" Will Be in Use in 2015," Gartner Inc., 2014

[6]. Nikhat Akhtar, Firoj Parwej, Dr. Yusuf Perwej, "A Perusal Of Big Data Classification And Hadoop Technology," International Transaction of Electrical and Computer Engineers System (ITECES), USA, ISSN (Print): 2373-1273 ISSN (Online): 2373- 1281, Vol. 4, No. 1, page 26-38, May 2017, DOI: 10.12691/iteces-4-1-4

[7]. Khadija Aziz, Dounia Zaidouni, Mostafa Bellafkih, "Real-time data analysis using Spark and Hadoop", 4th International Conference on Optimization and Applications (ICOA), IEEE, Mohammedia, Morocco , April 2018

[8]. Yusuf Perwej, Md. Husamuddin, Fokrul Alom Mazarbhuiya ,"An Extensive Investigate the MapReduce Technology", International Journal of Computer Sciences and Engineering (IJCSE), E-ISSN : 2347-2693, Volume-5, Issue-10, Page no. 218-225, Oct-2017, DOI : 10.26438/ijcse/v5i10.218225

[9]. Johnson Anumol, P.H. Havinash, Vince. Paul, Mr. Sankaranarayanan, "Big Data Processing Using Hadoop MapReduce Programming Model", International Journal of Computer Science and Information Technologies, vol. 6, no. 1, pp. 127-132, 2015

[10]. Tim Hegeman, Yong Guo, Mihai Capota, Bogdan Ghit, "Big Data in the Cloud: Enabling the Fourth Paradigm by Matching SMEs with Data Centers", 2nd ISO/IEC JTC 1 Study Group on Big Data, Amsterdam, 2014

[11]. Youssef Gahi, Mouhcine Guennoun, Hussein T. Mouftah ," Big Data Analytics: Security and privacy challenges", IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, June 2016

[12]. Firoj Parwej, Nikhat Akhtar, Yusuf Perwej, "A Close-Up View About Spark in Big Data Jurisdiction", International Journal of Engineering Research and Application (IJERA), ISSN: 2248-9622, Vol. 8, Issue 1, (Part -I1), pp.26-41 January 2018, DOI : 10.9790/9622-0801022641

[13]. Yusuf Perwej, "The Ambient Scrutinize of Scheduling Algorithms in Big Data Territory", International Journal of Advanced Research (IJAR), ISSN 2320-5407, Volume 6, Issue 3, PP 241- 258, March 2018, DOI : 10.21474/IJAR01/6672

[14]. D. Cutting, M. Cafarella, Apache Hadoop, http://hadoop.apache.org, accessed: 2019-06-10 (2006).

[15]. D. Borthakur, The Hadoop distributed file system: Architecture and design, Hadoop Project Website 11 (2007) (2007) 21.

[16]. HTCondor, Deploying High-throughput cluster using HTCondor over HDFS, HTCondor Manual, http://research.cs.wisc.edu/htcondor/ manual/v8.8/index.html, accessed: 2019-06-10 (2019).

[17]. M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, I. Stoica, Spark: Cluster computing with working sets, HotCloud 10 (10-10) (2010) 95.

[18]. P. Colombo, E. Ferrari, Access control in the era of big data: State of the art and research directions, in: Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies, ACM, 2018, pp. 185–192.

[19]. Apache Hadoop Project, HDFS federation, https://hadoop.apache. org/docs/current3/hadoop-project-dist/hadoop-hdfs/Federation. html, accessed: 2019-06-13.

[20]. Apache Hadoop Project, YARN federation, https://hadoop.apache. org/docs/current3/hadoop-yarn/hadoop-yarn-site/Federation. html, accessed: 2019-06-13.

[21]. K. Shvachko, H. Kuang, S. Radia, R. Chansler, The Hadoop distributed file system, in: Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on, Ieee, 2010, pp. 1–10.

[22]. Apache Hadoop, Apache Hadoop 3.x HDFS Federation Features, http://hadoop.apache.org/docs/r3.2.0/hadoop-project-dist/ hadoop-hdfs/Federation.html, accessed: 2019-06-10 (2019).