Review Article
Volume 10:9, 2021

Industrial Engineering & Management

ISSN: 2169-0316 Open Access

Hadoop Security and Privacy in Big Data

Hadiqa Amjad*, Nimra Jamil, Amna Azeem, and Saba Majeed

Department of Computer Science, Government College Women University Sialkot, Sialkot, Pakistan

Abstract

Big Data means the large amount of data. Data is rapidly increasing day by day. So, there's a big issue how to secure the large amount of data. Securing information is become a priority as data is a most important asset in today's world. In this paper we will discuss the challenges of Hadoop and provide the solutions for securing it. Some security challenges of Hadoop include Data Security, Network Security, Access Control Security etc. We have brief discussed about Hadoop framework, some security management modules and different approaches i.e. Kerberos technique, Bull Eye Algorithm, encryption techniques etc. in securing the Hadoop Ecosystem.

Keywords: Hadoop challenges • Hadoop security • HDFS Security

Introduction

Big data is basically the data that is big in size (in likely petabytes, exabytes or zettabytes) and is generated by various companies and many social media platforms. So, storing, securing, processing of that huge volume data is quite complex with using another databases software's. Also, many types of issues arise like managing issues that how to manage a huge volume data which is in structured, unstructured, semi-structured, Storage issues that where to hold such large data, Processing issues that how to process it and Security issues which is truly the main issue because large volume data can be hacked or attacked.

Literature Review

Hadoop (Highly Archived Distributed Object- Oriented Programming) is a platform which stores, manages, access, process, analyze and distribute big data in several nodes. It is an open source framework technology that gains data from big data clouds in different data formats and is in distributed fashion. It has one master node having metadata of client nodes and many slaves/client nodes having actual distributed data. Hadoop has two parts: Hadoop Distributed File System (HDFS) provides storage of data and Map Reduce provides analysis of data. HDFS is more reliable, scalable and distributed in Hadoop environment.

When data is transferring from one place to another, it is important to create such environment through which data can safely move or transfer another place, so in Hadoop, which storing and processing large amount of data provide different mechanisms while data is in motion Digest-MD5 and CRAM-MD5 before securing the data at motion the first step is to secure data at rest means data which is stored at a particular medium some techniques were proposed such as Encryption, Anonymization, data masking, and data erasure.

HDFS consists of 2 main nodes: Name Node and Data node. The NameNode also called the master node which is responsible for storing, managing and maintain the file system hierarchy as well as the metadata of all the files that are coming in Hadoop system. On the other hand, the data node is also called slave node which is to store and retrieve data as instructed by the Name Node. The data node stores multiple copies of each file that is storing in Hadoop file distributed system. The Name Node gives instruction to data node that store this data then data node store the data (Figure 1).

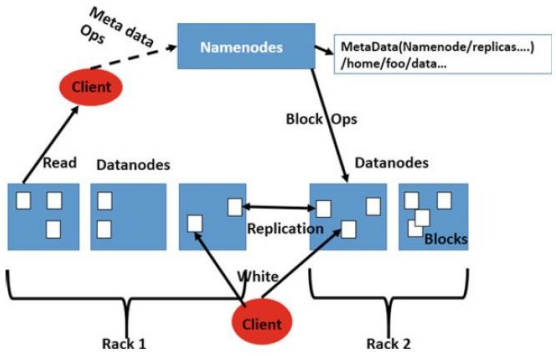

Figure 1. Hadoop architecture.

***Address to correspondence:** Dr Hadiqa Amjad, Department of Computer Science, Government College Women University Sialkot, Sialkot, Pakistan, Tel: 03248627497; E-mail: hadiqaamjad1@gmail.com*

Received: 29 July, 2021; **Accepted:** 12 August, 2021; **Published:** 19 August, 2021.

Security Challenegs in Hadoop

Network security

Internode communication: Hadoop framework has many nodes i.e. name nodes, data nodes, client nodes and data communication are performed b/w several nodes, so there need to secure internode communication as data is transferred over the network. Network security is mandatory.

User authentication problem: When client wants data transferring over the network, there needs to establish trust b/w client and server by authentication of user where client sends its password to server and server verifies its authenticity. Hadoop must enforce authentication mechanism.

Access security

Authorization & access control: Anybody who wants to access the data from HDFS must ensure its authenticity first to prevent sensitive data from hackers. Hadoop must enforce authorization rules that only authorized person will perform data retrieval. Observe the sensitive data and only allow authorized people to access and read or write the data.

Data security

Security of data in motion: When data transferring is over the network which is not secured and not in encrypted form, in this case data breaches may happen. Here, we need to provide end to end encryption over the network so that data is encrypted and protected from hacker's attack.

Security of data at rest: means data is stored in storage. The most sensitive data is stored in data-nodes. We also need to secure that sensitive data as this isn't in encrypted form by default. To safely storing the data in data nodes, we need to use such approach which enhances more security in data nodes or by using encryption techniques for preventing sensitive data from unauthorized access.

Secure and faster processing of data: Sometimes MapReduce tasks run on data, it takes time. Especially when data is in encrypted form and need to decrypt it for processing. So, need to make Hadoop framework be faster and secure processing of data.

System security

Checking in data breaches attacks: Analyze the data breaches attacks if any found in HDFS data blocks. Checking on data breaches will help to secure sensitive data from unauthorized access by taking some needed actions.

Data monitoring: Data should be monitored by some algorithm in routinely manner High, consistent data quality standards has set, and data monitoring keeps checks on these rules in our stored sensitive data.

Analyze the data leakage attacks: Analyze the data leakage attacks and find the attackers who steal the data. Evaluate the attacks found in Hadoop audit logs and in different layers.

Unavailability of namenode: Sometimes there's an issue of unavailability of NameNode so in this case data availability is also not possible in a securely way. So, we need to find a way to securely availability of the data from data nodes to authorized persons in this situation as well.

Hadoop Based Secured Solutions for Big Data

Network security

Kerberos mechanism: Kerberos was primarily designed by Steve Miller and Clifford Neuman published in 1980s. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. This is Network authentication protocol used to authenticate RPC connection without sharing password over the network. Token is used here for authenticating RPC connection. Connection is established by:

Remote procedure calls RPC: in-between client and Name node.

Block transfer: From Client to Data node.

TGT & service ticket: Here Ticket Granting Ticket and Service Ticket mechanism used for authenticating name node. It can be renewed if the task running for so long. Key distribution center (KDC) issue service ticket whenever it gets request from task.

It provides authentication without sharing password over the network, so password is protected by hackers. It's another advantage is if the attacker could access the ticket somehow, it can't be renewed.

The limitation of Kerberos is that after login it will not provide detailed level authentication. Its use is for single-user-client-system. But in any case, client will itself be multi-user system, that mechanism will not be suitable (Figure 2).

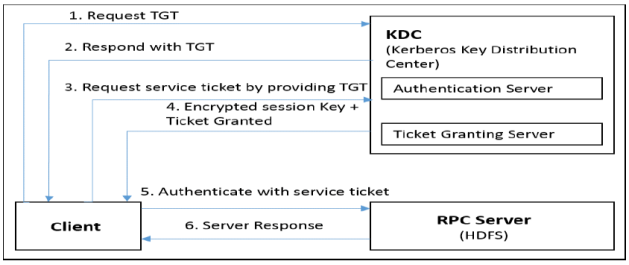

Figure 2. Authentication flow with Kerberos.

Access security

Block access token BAT: BAT was proposed by Brendan Eich, the inventor of the JavaScript programming language and co- founder of Mozilla and Firefox. Sometimes clients directly accessing the data from data node after accessing the data block ID from Name Node. So, in this case BAT is used to ensure that data blocks stored in data nodes can only be accessed by authorized users. Whenever any user wants to access any data node, it first accesses the name node to find out which data node holds the needed data blocks. Then name node issues the taken BAT which is used by data node to identify the authorized clients. Data blocks then verifies the token and issues Name Token by which it allows the NameNode to allow permission for securely access its data blocks.

Hashing techniques SHA-256: Hash-256 Algorithm was designed by National Security Agency (NSA). This mechanism is implemented b/w user and name node and ensure authentication and manages data nodes. The user authenticates himself to NameNode by sending a hash function. First user sends hash function to get access of data. name node also produces the hash function. After this both hash function compares, if both are correct then data can be accessed successfully.

Data security

Random encryption techniques i.e. RSA, rijndael, AES and RC6: Rijndael and AES algorithm proposed by Belgian cryptographers, Vincent Rijmen and Joan Daemen. RSA algorithm is proposed by Ron Rivest, Adi Shamir, and Leonard Adelman. RC6 is by des Ron Rivest, Matt Robshaw, Ray Sidney, Yiqun Lisa Yinigned in 1998. These Algorithms are used to encrypt sensitive data and secured our data from unauthorized access. Map reduce is used for encryption/decryption processes in these techniques. The encryptions key is highly protected from unauthorized access. In these encryption techniques, 128-bit size of cipher blocks are used, and different key sizes are used to encrypt or decrypt the data. Through these technique/algorithms' hackers will not break the cipher text and not access the private data.

Eencryption zone: It is a technology employed by Microsoft, IBM and Oracle to encrypt database files. Microsoft offers TDE as part of its Microsoft SQL Server 2008, 2008 R2, 2012, 2014 and 2016. It is used for transparent encryption, an abstraction to HDFS. A special directory whose data will be encrypted transparently for write and decrypted transparently for read. When encryption zone is formed, its key is generated. Every file in encryption zone is specified by its own unique Data Encryption Key (DEK). HDFS only handles an Encrypted Data Encryption Key (EDEK). Clients decrypt this EDEK and then they could use DEK for read/write operation. DEK and other encrypted keys in Hadoop framework is handled by Hadoop Key Management Server (KMS). KMS is responsible for client access to required key, generating new key, decrypting the key for client access etc.

As it helps in encrypting data from hackers, same as it has some limitations. It is not suitable for data processing. Whenever map reduce task run on encryption zone data, complete data file is first decrypted for processing and after running again encrypting the data. So, a lot of time is wasting in this process which badly affects its performance.

Hadoop's remote procedure call using SASL framework (Encryption for data in motion): is used to provide user authentication b/w name node and job tracker services. SASL framework can be used to encrypt the data when data transferring needed. So that it protects the data is in motion and ensures communication is encrypted and secured. Kerberos authenticate users within SASL framework.

ChaCha20 algorithm: ChaCha20 is a stream cipher developed by Daniel J. Bernstein. Its original design expands a 256-bit key into 2^64 randomly accessible streams, each containing 2^64 randomly accessible 64-byte (512 bits) blocks. It is a variant of Salsa20 with better diffusion. Here, Map reduce task doesn't need to decrypt the complete file for processing. In this framework HDFS encryption zone is eliminates. Clients splits complete data file into blocks of size and encrypts each individual block. Then map reduce process these individual blocks and decryption is applied during the map phase for a MapReduce job by getting decryption key to decrypt the file. It provides three-dimensional security, better speed, memory efficient, easy to implement, flexible and secure. In this method, each individual encrypted block will be decrypted individually as well. For analysis of this algorithm, Mahout environment is used over the clustered data.

Moreover, it doesn't provide guarantee of authenticity of the data which is decrypt. Hacker can easily access the data that is in motion. For protection, Message Authentication Code can be used for authenticating or you can use ChaCha20_Poly1305.

System security

H.Bull Eye Approach: Bull Eye Algorithms was designed by MIT to provide the solution of Network security problems. These Algorithms used to secure the sensitive and private data and provides security from node to node i.e. monitoring all sensitive data in 360°. It is implemented in Data node of rack 1 and checks to ensure our sensitive data are stored safe in block. It only allows authorized person to access the data nodes and only allow particular client to store in blocks of data nodes.

This algorithm is also used to fill up the gap b/w real and replicated data nodes and checks the relation b/w two racks.

This algorithm ensures only authorized person read or write on the data nodes. Besides data node rack 1, it also implemented in rack 2 for ensuring high level security inside the data nodes as well. It is analyzing the data in 360° before and after entering the data into the blocks in rack 1 and 2 both.

Not only stored data, it also be implemented in encrypted data in order to provide more security. It constantly checks for attacks, breaches etc. if any in data node blocks.

Automation detection algorithm: Automatic Detection Algorithm are designed to detect the da ta leakage attacks in Hadoop nodes. Leakage can be in the Application layer and operating system layer. In OS layer Automatic detection Algorithm work in four dimensions i.e.

Abnormal Directory (AD), Abnormal user (AU), Abnormal operation (AO) and Block proportion (BP). Hadoop have fixed directory to save files if any directory is out of range then it means that there happened an attack. And then it finds the suspicious block that contain abnormal directory and calculate suspicious block proportion.it also detect the Abnormal user. If any of these four dimensions gives warning, it means there happened an attack. And then investigator's start to investigate the problem according to warning message. In Application layer investigation based on Hadoop logs i.e. NameNode, data node.

Namenode approach: is used to overcome the problem of unavailability of NameNode, we have an approach called NameNode approach in which we make two NameNode servers. Two name nodes run on the same distributed network provided by Name Node Security Enhance (NNSE). One act as Master node and other act as a slave node.

In any case if master nodes crash or any error occurs, so HDFS administrator can get permission by NNSE to get access of data by slave node so in this case data availability is secure and time saving. It also helps to reduce server crashes.

One more thing to be kept in mind is if both name nodes acts as a master node, it'll create more problems, risks arises, performance affects and data accessing will be unsecure.

In future, Vital Configuration will be used by replicating name nodes with NNSE which provides secure data accessing to clients (Table 1).

Sr. No	Type	Issues / Challenges	Model	Short Description about Model	Significance	Limitations / Future Work
1	Network Security	User Authentication Problem		Network Authentication Protocol used to authenticate RPC connection by using TGT & Service Ticket.	Provides authentication without sharing password over the network.	In future, it will deploy in large Hadoop clusters.
2		Secured Internode Communication	Kerberos Mechanism			
3	Access Security	Authorization & Access Control	Block Access Token BAT	It is issued by namenode and used by data node to authenticate users.	Ensures data blocks can only be accessed by authorized users	it's work flow is quite complex.
			Hashing Technique SHA-256	Ensures authentication of users by providing hash function by NameNode.	This ensures authentication first to access name node and data nodes.	It isn't a secure password hashing algorithm.
4	Data Security	Security of data at rest	Encryption Techniques i.e. RSA, Rijndael, AES and RC6	Used to encrypt data & MapReduce is executed encryption/ decryption processes.	Map reduce helps in speed up encryption / decryption processes.	Combinations of techniques will be used to get more secure data storage & retrieval.
			Encryption Zone (transparent encryption)	Special directory whose data will be encrypted/ decrypted transparently for read/ write.	Provides transparent encryption to HDFS will help for securely storing the data.	Encryption zone is not being utilizing for processing of data.
5		Security of data in motion	RPC using SASL framework	SASL framework can be used to encrypt the data when data transferring needed.	Protects the data in motion and ensures communication is encrypted and secured.	Further work will go on in improving its speed.
6		Secure Processing of data	ChaCha20 algorithm	Splits complete data file into blocks of size and encryption / decryption is applied in each individual block.	Provides 3-dimensional security, better speed, memory efficient, flexible and secure.	Doesn't provide guarantee of authenticity of the data which is decrypt.
7	System Security	Checking in Data Breaches Attacks	Bull Eye Algorithm	Provides security from node to node i.e. monitoring all sensitive data in 360°.	It constantly checks for attacks, breaches etc. in data node blocks.	Will work on its efficiency more.
8		Monitoring Sensitive Data				
9		Data leakage Attacks	Automatic Detection Algorithm	Designed to detect data leakage attacks.	It protects the OS layer from data leakage & find the suspicious blocks very efficiently.	It only works in OS layer. More work will make to improve its efficiency.
10		Unavailability of name nodes	NameNode Approach	Master & slave name nodes run on the same distributed network provided by NNSE.	Administrator can get permission by NNSE to get access of data by slave node if any error occurs.	Vital Configuration will be used by replicating name nodes with NNSE.

Table 1. Security challenges and solutions.

Some Security Management Modules in Hadoop

Authentication management module

This module is based on Kerberos authentication that provide authentication services for big data. This module is divided into

Kerberos authentication management and user information management. The Kerberos authentication management sub-module manage the services of Kerberos authentication in the big data platform and its functions are startup and shutdown of the Kerberos service, Kerberos service parameter configuration, Kerberos authentication key management, and Kerberos ticket management and the purpose of user information management sub-module is to manages the user's information authentication in the big data platform. The Function of User information management is creating users, view user's information, delete users and modify user's authentication

Static data encryption management module

This module uses Transparent Encryption i.e. Encryption zone. It is responsible for start-up and shutdown of transparent cryptographic services, the generation of encryption keys, selection of static data encryption algorithms, configuration of KMS, and creation of transparent encryption zones.

Apache sentry

Apache Sentry is started at Cloudera and Entered incubation in 2013 and its Growing community is Committers from Cloudera, IBM, Intel, Oracle, and three releases from incubation. It is widely adopted by industry and it is part of multiple commercial Hadoop distros. Apache sentry is the open source tool of Cloudera. It is an authorization module provides support for Role Based Access Control RBAC (Support for role templates to manage authorization for a large set of users and data objects), fine-grained authorization (Permissions on object hierarchies. E.g., Database, Table, Columns), and multi-tenant administration. (Ability to delegate admin responsibilities for a subset of resources).

Apache rhino

Apache Rhino is an effort of Cloudera, Intel and Hadoop community to bring a comprehensive security framework for data protection. To catalyze the development of a comprehensive security framework for data protection in Apache Hadoop, Intel launched Project Rhino in early 2013 as an initiative with several broad objectives: It provide encryption with hardware-enhanced performance and support enterprise-grade authentication and single sign-on for Hadoop service. It ensures consistent auditing across essential Apache Hadoop components. It's aims is to provide overall security solution for data in entire Hadoop ecosystem. It provides a framework which is crypto codec that offers block level encryption of data stored in Hadoop (Table 2).

Sr. no	Typ e	Management Perspective	Module	Short Description about Module	Significance	Future Work
1	Security Management Modules	Authentication management	Authentication management module	Module is divided into Kerberos authentication management and user informati on.	Manages authentication services of Kerberos and user's information.	Keep working on security management system in Hadoop platforms
2		Encryption Management	Static data encryption management module	Uses transparent encryption technology.	Through Encryption Hackers cannot access Data.	
3		Authorization management	Apache Sentry	Provide role-based access control RBAC to restricting access by unauthorized access.	It provides support for fine-grained authorization, multi-tenant administration	
4		Overall Security Management	Apache Rhino	Provides overall security in Hadoop ecosystem.	Provides token-based authentication, block level encryption, SSO solution.	

Table 2. Description about modules.

Discussions

This paper represents many security challenges and threats arises in Hadoop framework and trying to reduce them by using various approaches that are discussed above. Security issues associated with security of data in motion, security of data at rest and authentication and authorization issues over a network, analyzing data breaches and leakage attacks, unavailability of NameNode that causes problem in securely accessing data from data nodes, slow processing have been highlighted and to overcome these issues some approaches used. Kerberos is a network authentication protocol used to access data block only by authorized users. SASL framework encrypt the data that is travelled over a network to prevent data leakage from unauthorized access. Bull eye algorithm is used to provide security from node to node and monitor the data in 360°. NameNode approach is designed to overcome the availability of NameNode in which we make two NameNode servers and helps to prevents from server crashes. BAT is used to ensure that data blocks stored in data nodes can only be accessed by authorized users. Hashing techniques SHA-256 is implemented b/w client and name node and ensures authentication and manages data nodes. The user authenticates himself to Name Node by sending a hash function. Random encryption techniques i.e. RSA, Rijndael, AES and RC6 are used to encrypt the data and secured our data from unauthorized access. Encryption zone is a special directory whose data will be encrypted transparently for write and decrypted transparently for read. Automatic Detection Algorithm are designed to detect the data leakage attacks in Hadoop nodes. Next section we have discussed some security management modules Authentication Management Module, Static Data Encryption Management Module, Apache Sentry,

Apache Rhino used for authentication, authorization, encryption/ decryption and overall security management.

Although all approaches doing their best job in reducing security threats and making Hadoop a well secure framework, still we can't say security threats are eliminated completely. Cybercrimes are increasing day by day and we need to make our best security measures to tackle them. All approaches are designed to tackle with some specific challenge so our recommendation to mark any one approach best out of all is quite tricky, yet we recommend ChaCha20 algorithm as it provides three-dimensional security, better speed, memory efficient ,easy to implement, and flexible but it doesn't provide guarantee of authenticity. For authentication we recommend Kerberos technique because Kerberos ensures authenticity of users. Better to use combinations of these techniques will give more assurity about security perspective.

Conclusion

This paper highlights what security issues arises in Hadoop platform and how to tackle them in an efficient way. Different challenges in security perspective are discussed and how to overcome them by using different approaches to keep Hadoop a secure framework for storing, accessing, processing large volume of big data. Discussed approaches are implemented in HDFS layer to make it secure and reliable at the same time keeping the performance standards. Combinations of these approaches can also be used. All these approaches give their best part in improving security but where they have many advantages, some limitations can't be ignored as well.

In future, these approaches will try to be implemented in other layers of Hadoop. Security and safety of large volume big data is trending issue and to be work more in future. So new approaches, new mechanisms will be developed with time along with the improvement in these approaches to make Hadoop more secure framework than before. Furthermore, development of new systems is also be under consideration in order to fully stop cybercrimes.

References

1. Saraladevi, B, N Pazhaniraja, P Victer Paul, and MS Saleem Basha, et al. "Big Data and Hadoop-A Study in Security Perspective." *Proc Comput Sci 50 (2015): 596-601.*
2. Terzi, Duygu Sinanc, Ramazan Terzi, and Seref Sagiroglu. "A Survey on Security and Privacy Issues In Big Data." In 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), IEEE, (2015).
3. Balaraju, J, and P V R D Prasada Rao. "Recent Advances In Big Data Storage and Security Schemas Of HDFS: A Survey." *J Eng Technol* 118 (2018): 132-138.
4. Behera, Manoranjan, and Akhtar Rasool. "Big Data Security Threats and Prevention Measures in Cloud and Hadoop." *Data Manag Analy Innov* (2019): 143-156.
5. Adluru, Pradeep, Srikari Sindhoori Datla, and Xiaowen Zhang. "Hadoop Eco System for Big Data Security and Privacy." In 2015 Long Island Systems, Applications and Technology, IEEE, (2015).
6. Fu, Xiao, Yun Gao, Bin Luo, and Xiaojiang Du, et al. "Security Threats to Hadoop: Data Leakage Attacks and Investigation." *IEEE Network* 31 (2017): 67-71.
7. Gaddam, Ajit. "Securing Your Big Data Environment." Black Hat USA 2015. (2015).

How to cite this article: Amjad, Hadiqa, Jamil Nimra, Azeem Amna, and Majeed Saba. "Hadoop Security and Privacy in Big Data ." *Indu Eng Manag*10 (2021) : 3332