International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/IJSRCSEIT

Security and Challenges in Implementation of Hadoop Technology in Cloud

Nuzhat Shabir[*1], **Manoj Kumar Srivastava**[2]
[*1] M.Tech (Scholar), CSE Depatment Desh Bhagat University, Mandi Gobindgarh, Punjab, India
[2] Assistant Professor, CSE Department, Desh Bhagat University, Mandi Gobindgarh, Punjab, India

ABSTRACT

Article Info
Volume 7, Issue 4
Page Number: 252-261

Publication Issue :
July-August-2021

Article History
Accepted : 10 July 2021
Published : 18 July 2021

In this Cloud Computing era, major area of worry today is managing terrific increase of over flown and exponentially growing data. The security particularly for organizing data in database. The rising exponential Internet of Things (IoT) data has led to several governance issues for government and private organization. Threats to security have compelled corporate and public sector businesses to build their own Hadoop-based cloud storage infrastructure. It builds numerous clusters of computers and efficiently coordinates work among them under the Apache Hadoop architecture. HDFS (Hadoop Distributed File System) and Map Reduce (Hadoop Map Reduce) are two key Hadoop components. HDFS is Hadoop's principal storage system. It allows for accurate and quick calculations. HDFS enables diverse user applications executing for end user with high-availability rich data set. Map Reduce framework software for analysing vast amount of data and transform into the required output. Here it examines HDFS architecture, as well as its numerous functionalities, such as analytical and security capabilities.

Keywords : Cloud Clusters, Security in Hadoop, HDFS, Map Reduce, Cloud Computing

I. INTRODUCTION

The Open source Hadoop technology is framework for storing organized, unstructured, semistructured, and quasistructured data; such data is often referred to as voluminous big data. It uses data analytics to provide useful results. Extract, Transform, and Load is a standard approach for working massive volumes of data. gathering information from a range of sources, to satisfy analytical needs, and feeding it into the right systems to provide useful information. It has a number of advantages for charity organizations and government. Hadoop is a highly scalable platform developed in JAVA, which consists of distributed File system that allows multiple concurrent jobs to run on multiple servers splitting and transferring data and files between different nodes. It is efficient to process or recover the stored data without any delay in case of failure of any node. At the same time chances of fraudulence increases while processing or storing information in HDFS. Due to various big data issues with respect to management, storage, processing and security, it is necessary to deal with all individually. Companies are turning to Hadoop technology to deploy digitalization in their organisations and to reap all of the benefits. Hadoop is a JAVA-based, highly scalable platform that includes a distributed file system that allows several concurrent processes to execute on several servers while dividing and distributing data and files between nodes. In the event of a node failure, it is efficient to process or restore the stored data without delay. At the same time, while processing or storing data in HDFS, the risk of fraud grows. Due to a slew of massive data storage, processing, and security challenges, management, It is vital to deal with each separately. [8].

Hadoop's challenges for User:

It has several obstacles that must be resolved in order for every organization to be able to rely on it and store ever-increasing amounts of data in a secure and reliable manner. The following challenges are:

1) Constant data development: The Hadoop technology clusters also have to be adjust the data as is always exponential and expanding. Their ecosystem comprises of complicated software that continually changes according to demand and necessary for data sets to be maintained. The current scenario lacks standards or guidelines that can offer the optimum platform for safe operation..
2) Don't set. Working Platform: It is up to the end users to pick according to their needs, however the Hadoop technology has no predefined platform. Simultaneously, end users may not have good hardware expertise to adequately resolve the problem

The issue to be discussed

The Hadoop based architecture of cloud storage is used by all current and increasing business and governmental organisations. The design of Hadoop stores all critical and personal information. In several nodes, clusters or servers it saves sensitive information in the form of individual files. It leverages so many database for Hadoop such as Pig, HBase and Mahout technologies to better analyse data. Most government and private business companies worry the retention of their Hadoop data [13]. Hadoop does not come with a security by default, which leads to a slew of security risks such as personal information abuse and fraud. While Hadoop implementation Hadoop, the security measures should be established in such a way that the user should be allowed to use data only if authenticated, and that there is no counterfeit or misused information.

II. REVIEW OF LITERATURE

Map Reduce, according to J. Zhao, L. Wang, J. Tao, J. Chen, W. Sun, and R. Ranjan, is the finest programming methodology for large-scaled information-based applications[1]. A Hadoop-based system uses map-reduce programming to run across several clusters. Hadoop's client validation and occupation accommodation mechanism, which is built for a single group, is reused by G-hadoop. Based on the SSL protocol and open cryptography, they presented a Hadoop paradigm of security. With a sign-on method, this structure it supports the existing G-Hadoop execution's client confirmation and employment accommodation procedure.

For data security in the HDFS context, V. Kadre and Sushil Chaturvedi proposed the AES-MR encryption approach. One of the most effective ways for encrypting data is the AES encryption technique. It runs in the background. The XTS mode provides for parallelization and pipe lining, allowing for the last piece of data to be added. [2]

Monika Kumari, Dr. Sanjay Tyagi presented a three-tiered security strategy for Hadoop data management For communication with verified users, this technique provides a secure tunnel-based transmission. The RSA technique provides one-time authentication, and layer SSL is engaged for access services of Hadoop. To provide users access to the public areas, RSA-based authentication is used. The intermediate layer implements security, which is separated into three sections: secure data management, authorization, and secure session. [7]

Laxman Gaikwad et al. The Network Enhancement of Security for Hadoop Clusters is provided by introducing the authentication automation using delegation tokens and proposing updated security models utilising the Access Control on Role based with discussion of web authetication improvements of Hadoop Clusters' users.[3]

Veiga et al. [14] to suit our multidimensional comparison's performance component This decision was taken due to the fact that a broad and entire evaluation not just a piece. This would not be able as extensive and rigorous performance comparison

There is no doubt that a lot of study has been done on the performance of these distributed computing engines. Multidimensional comparisons, on the other hand, have much fewer options, particularly those focusing on non-performance comparisons like usability.

Mehta et al. offer a research evaluating the applicability and performance of big data processing for scientific image analysis processes. [15]. This research also gives a qualitative overview of various system, including the simplicity of use and the general difficulty in implementation, although based on code measurement lines and obstacles in installation – not totally in line with the utility of the systems..

Richter et al. offer a multidimensional comparison of Apache Hadoop MapReduce, Apache Storm, and Spark in the context of several distributed computing techniques like as k-means and linear regression using some of their higher level APIs, such as MlLib for Spark [16]. The parameters of the comparison include four performance or "capacity" related metrics: speed, fault tolerance, scalability and extensibility, and usability. However, the usability dimension appears to be mostly focused on static analysis, such as available interface features or programming language compatibility, with little information on the ease of use or human contact element, as well as how the final results were calculated. This is again another nice performance comparison, although the usability factor is lacking in depth.

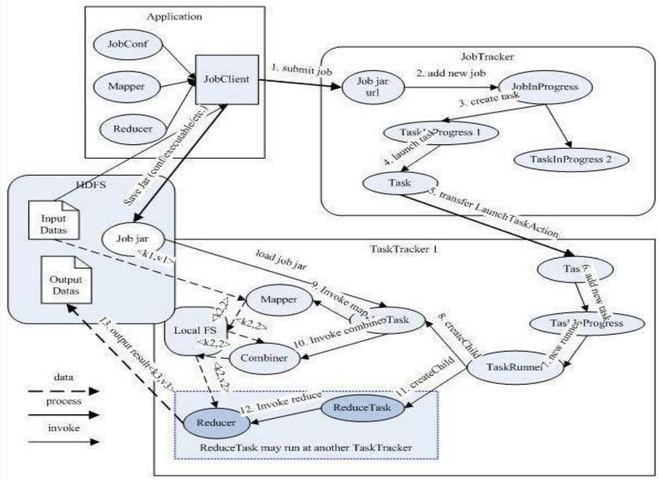
Galilee et al. offer a poster that compares Hadoop MapReduce, Spark, and Flink in terms of performance, usability, and practicality [17]. Although it is nice to see an emphasis on these non-performance related characteristics, given the nature of the publication, there is a lack of transparency in how the ratings were collected, as well as concerns about the subjectivity of having been completed solely from the perspective of a single researcher.

It can see that performance comparisons abound, and that what we've discovered will suffice in accommodating the performance aspect of our multidimensional comparison – specifically, the data from Veiga et al. [14], which provides both the system coverage we need and a level of quality that we consider reliable. Now we'll turn our attention to the crucial criteria of usability and practicality, which were noticeably absent in the preceding articles. Potential users may struggle to correctly and effectively determine the applicability of these solutions to their use cases if this information is not provided. This is especially problematic for people who do not have a strong technical background, as they perhaps would not be able to compare and judge these factors themselves

An open source version Hadoop of Google's graphs computation model that is part of the Apache fund account component. It's simple to programme and operate large-scale data processing. HDFS (Hadoop Distributed File System) and MapReduce are two of the most important components. [18]

HDFS: It's a distributed file system that stores large files with streaming data access patterns and operates in a managers-workers mode, which means there's a manager named Name Node and a lot of workers named Nodes (Data Nodes). Manager Node is in charge of the file system tree, as well as the trees in all directories and files (Name Node). Each file in each block of Data Node data is tracked by a worker node(Data Node), which is a cluster node. MapReduce is a technique for reducing the size. In the MapReduce work process, the Map and Reduce stages are separated. A Map function is used to convert a set of keys into a new set of key-value pairs for mapping. It can also be used for the Reduce function. MapReduce has several components: a submission and startup structure, task allocation, job execution, and job completion. [19]

The Job Client submits it first, and then the job details are uploaded to the Job Tracker. The Task Tracker is the heart of the Map-Reduce system, and it must communicate with the machine cluster to handle job failures and resume operations, as well as regulate which programmes be run on which computers. In MapReduce, TaskTracker is a component of each machine. It's made to keep an eye on their computers' resources. TaskTracker keeps track of the machine's present condition and does monitoring jobs. TaskTracker provides the information through JobTracker's pulse. JobTracker will use this information to allocate a new job that has been submitted to a machine.



Hadoop YARN

Hadoop YARN's background

The architecture of MapReduce is clear and straightforward. It also received a large number of successful cases within the first few years of its introduction, as well as industry-wide support and confirmation. However, when the cluster scales and the workload grows, the original structure of the problem appears, and the basic challenge is as follows

- The JobTracker focuses on the map-reduce, there is one failure point;
- The JobTracker did most job, resulting in high use of resources. If the map – reduces work considerably, it can produce a large overall memory, increases the risk of failures by JobTracker, furthermore the industry is typically resuming the old Hadoop map – reducing the host limit that supported;
- The framework does not include CPU/memory utilisation in TaskTracker. When the work is programmed for two huge memory usage, OOM is simple to appear;

The Hadoop Framework

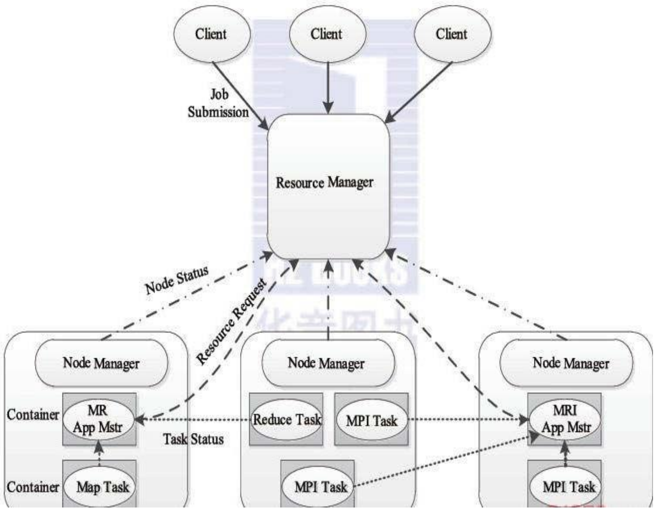


Figure 2. Framework Architecture of YARN

- Resource Manager: The global resource manager (RM) who manages and allocates resources throughout the whole system. The Scheduler and the application Manager are two components of global resource Manager, RM.
- ApplicationMaster: Every request has one ApplicationManager. The primary characteristics are: to Discuss with RM Resource Scheduler, tasks under the assignment further allocated, Contact NM to start/stop the job and monitor all functions;
- NodeManager : NM is a Task Manager for every resource node. So, this node will report on resource use and the functioning of each container to the RM periodically. Instead, from AM start/stop and from other requests, it receives and handles the container;
- •Container: The YARN resource is container abstraction. The multi-dominant resources of a node such as memory space , CPU, hard disk, networking, and so on. are encapsulated.

The process of Hadoop YARN

YARN shall execute the app using the following procedures when the user submits an application to YARN:

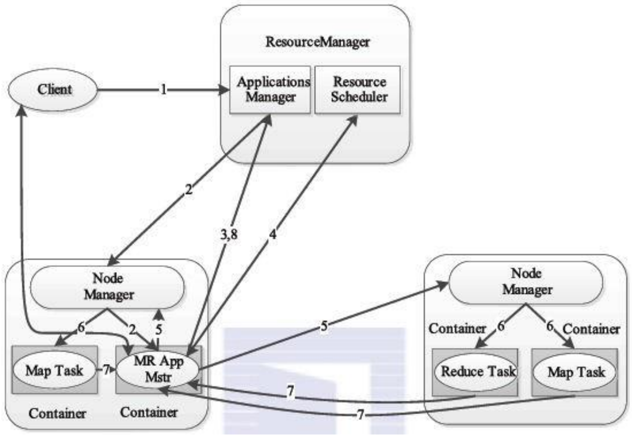


Figure 3. YARN working principles

Users submit YARN applications, including ApplicationMaster, Startup- Commands for ApplicationMaster, User Program;

- The first container for this application is distributed by the ResourceManager. It contacts the Node - Manager, which has to be start the application in the ApplicationMaster Container;
- ApplicationMaster register with the ResourceManager firstly. The user can view application running state directly through the ResourceManager;
- ApplicationMaster adopt the way of polling by RPC protocol applied to the ResourceManager for and the resources;
- Each task reports to the ApplicationMaster through the RPC Protocol its own status and progress..
- Each task's state may be determined by ApplicationMaster.;
- ApplicationMaster logs out and closes down after the conclusion of an application..

III. Comparison and Methodology

It is difficult to compare several technologies while taking into account different criteria and communicating the results to people with varying technical expertise. Reduced biases between technologies must be taken into account. The items selected must pique the target audience's attention. The method of communication must be acceptable for readers who know the technical details of the comparison and those who may not be able to comprehend such information and prefer a more simplified approach. Many number of additional factors to consider, while a strategy or theoretical framework is essential prior to conducting a comparison.

We were unable to find any contemporary methodologies or frameworks to employ in our comparison, though., concepts or procedures to utilize in the building of a new methodology. As a result, we propose an early suggestion for a multidimensional software comparison approach to give structure while taking into account all of these requirements. The approach guides the planning, interpretation, and presentation of comparison findings in a style that is acceptable in a variety of situations and for a variety of audiences for comparison. Not a very controlled and precise strategy in this regard, as doing so would drastically restrict its use. Instead, it seeks to assist the process of decision-making that takes place prior to the start of comparisons, with the goal of improving comparison reliability and utility.

In the next part, the proposed methodology will be detailed and justified. The results section will next employ the provided approach to undertake a comparison of Apache Hadoop MapReduce, accounting for each system's practicality, performance, and usability.

IV. Methodology

The methodology presented at the beginning of the chapter will be discussed here. Initially, we give a high-level summary of the method, outlining it as a series of steps and elaborating on the practical consequences for readers of the resulting correlations - useful as a quick reference during use. The methodology's design and our justifications for each stage will be explained after that.

Model for Security Implementation Proposed: In the Security layer, I suggest employing multiple ways to accomplish the security aspects specified in this article. The suggested model is as follows:

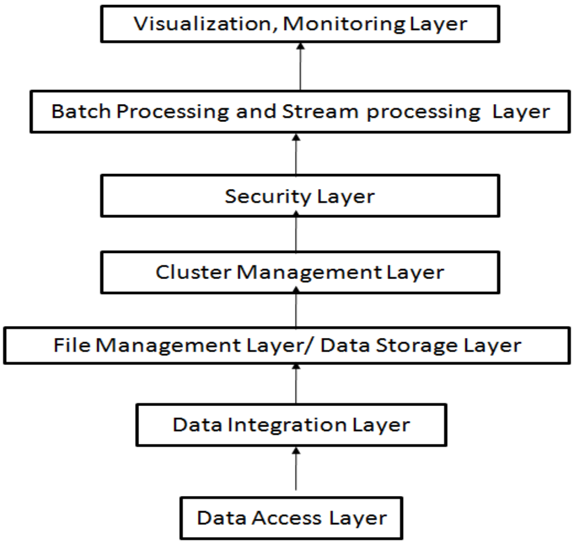

Visualization, Monitoring Layer
Batch Processing and Stream processing Layer
Security Layer
Cluster Management Layer
File Management Layer/ Data Storage Layer
Data Integration Layer
Data Access Layer

Fig 6. Proposed security implementation in Big Data

V. Design and Justification

It is noticed while designing a technique of comparison, it was easy for the particular approach, such that it would only be effective for our study and other very comparisons for comparable – say, of distributed computing engines. Hence, instead of proposing a technique, rather constructing one for use in our present study.

However, the situation described throughout this work – scientists lacking clear, reliable comparisons to support their selection of a distributed system – is one that could benefit from these comparisons, there are undoubtedly many other contexts that would benefit from a continuous stream of comparisons. So, rather than focusing just on our environment, we try to give assistance in multidimensional comparisons to scholars interested in going down that route in other settings as well, because we saw firsthand how little guidance available.

However, system comparisons are usually highly particular. Distributed systems, for example, scalability and fault tolerance must address, though this is not possible in programming languages. One of the major metrics in comparing compression of image libraries is the size of file with images generated. Systems for analyzing streams of video have to consider not just throughput but latency of result. This goes on, and we are using it to emphasize how tough it is to create a technique that can be applied to all of these instances.

As a result, the technique is on the higher end of the spectrum. Its goal is to aid the researchers doing the comparison in their decision-making process, such as reducing errors that might endanger the integrity of performance metrics. – For instance, lacking empathy with the audience in the comparison (which we have virtually done) – while yet allowing them the strength to adapt their system to their needs.

With a variety of scenarios, we feel other important matter should be addressed, that the approach should examine is the various audiences and their degree of contextual awareness, which may and should be taken into account when performing a comparison.

VI. Implementation in Security Layer

The AES algorithm outperforms the DES, according to RSA. However, one downside of the AES method is sharing a key. There seems insecure way to hand out the key. When we compress a huge file, we lose data as well. These algorithms' key number, block, assurance rate, and execution time all created security concerns. [9].

Features proposed to address Large Data Security problem:

1) Authorization: It guarantees that safe administrator passwords so that all users of application who need access to the specific cluster must be authenticated. Every one has a unique set of skills accessing password for example administrator roles developers role, and users should be segregated.
2) Privacy Sharing: It has variety of integration types available. The aim behind big data security analytics is to use various data mining techniques to accumulate more vital or responsive data in a huddle inside a cluster.
3) Authentication of Node: Some shield against combining undesirable apps and nodes to a big data cluster, particularly in cloud computing and virtual machine environments where copying a VM image and starting an instance is straightforward. Rogue nodes cannot issue queries or obtain data copies thanks to Kerberos tools. [10].
4) Data Encryption: This is a critical feature for making massive data more safe and accessible only to administrators. File/OS level encryption is preferred since it grows as nodes are added and is clear to processes of NOSQL.
5) Key Management: The most critical aspect of data encryption is key security. Any everlasting key management system should use safe keys and, if feasible, assist in key validation.
6) Logging: Hadoop clusters incorporates logging. It protects all other network devices and programmes, and it is advised that users utilize the inbuilt logging or one of the commercial logging solutions available to record a system events.

Table 2 : Results of the Experiments.

Item	Small	Medium	Large
Cluster provisioning	160 sec.	180 sec.	190 sec.
Size of the dataset (Avro)	400 MB	4 GB	18 GB
Size of the dataset RDBMS	-	-	76 GB
Cluster data analysis	8 sec.	80 sec.	150 sec.
RDBMS data analysis	6 sec.	90 sec.	201 sec.
NoSQL query by key	0.03 sec.	0.06 sec.	0.1 sec.
RDBMS query by key	0.01 sec.	0.02 sec.	0.04 sec.

VII. RESULT

In this section, we examine Apache Hadoop, MapReduce's usability, performance, and practicality in data visualization, using methods stated before for evaluating their usability, performance, and practicality in data storage science. Because whole description of dimensions offered at bottom tier in evaluation of results, as the approach utilized in a series of steps as stated by the methodology, but with the individual comparison and measurement findings combined into one.

VIII. CONCLUSION

The Hadoop usable technology, their mechanism, the advantages of file system HDFS, with Map Reduce technique in this paper. Businesses are turning to big-data management tools to deal with the influx of data. In this case, it is necessary to investigate a variety of technological issues. as well as their security implications. The suggested approach adds a new layer called Security, which includes an AES compression and implementation with it.

IX. REFERENCES

[1]. Zhao J., Wang L., Tao J.,Chen J., Sun W., Ranjan R., et al., "A security framework in G-Hadoop for big data com- puting across distributed Cloud data centres," Journal of Computer and System Sciences, vol. 80, pp.994-1007, 2014

[2]. Kadre Viplove , Chaturvedi Sushil , "AES – MR: A Novel Encryption Scheme for securing Data in HDFS Environ- ment using MapReduce http://www.ijcaonline.org/research/volume129/number12 /kadre-2015-ijca 906994.pdf, International Journal of Computer Applications (0975 – 8887) Volume 129 – No.12, November2015.

[3]. Gaikwad Rajesh Laxman , Prof. Dhananjay M Dakhane and Prof. Ravindra L Pardhi," Network Security Enhancement in Hadoop Clusters", IJAIEM ,Volume 2, Issue 3, March 2013 ISSN 2319 – 4847.

[4]. Saraladevia B.,Pazhanirajaa N., Victer Paula, Saleem Bashab, Dhavachelvanc P.," Big Data and Hadoop-A Study in Security Perspective", 2nd International Symposium on Big Data and Cloud Computing (ISBCC'15).

[5]. Karthik D, Manjunath T N, Srinivas K," A View on Data Security System for Cloud on Hadoop Framework", http://research.ijcaonline.org/nckite2015/number3/nckite2 661.pdf, International Journal of Computer Applications (0975 – 8887) National Conference on Knowledge, Inno- vation in Technology and Engineering (NCKITE 2015).

[6]. Vinit G. Savant," Approaches to Solve Big Data Security Issues and Comparative Study of Cryptographic Algorithms for Data Encryption ",http://ijicar.com/wp-

content/uploads/2015/04/RJ010106.pdf, Volume 1 : Issue 1 International Journal of Integrated Computer Applica- tions & Research (ijicar) idin rJ010106 ISSN 2395-4310 2015 © IJICAR http://ijicar.com.

[7]. Monika Kumari ,Dr.Sanjay Tyagi ,"A Three Layered Se- curity Model for Data Management in Hadoop Environ- ment " https://www.ijarcsse.com/docs/papers/Volume_4 / 6_June2014/V4I6-0105.pdf , Volume 4, Issue 6, June 2014 ISSN: 2277 128X International Journal of Advanced Re- search in Computer Science and Software Engineering Research Paper Available online at: www.ijarcsse.com.

[8]. B. Saraladevia, N. Pazhanirajaa, P. Victer Paula, M.S. Saleem Bashab, P. Dhavachelvanc ," Big Data and Hadoop-A Study in Security Perspective ",http://www.sciencedirect.com/science/article/pii/S187705 091500592X,2nd International Symposium on Big Data and Cloud Computing (ISBCC'15).

[9]. Ms. Chetana Girish Gorakh, Dr. Kishor M. Dhole,"A Re- view on Security Approach in Big Data",http://www.iosrjournals.org/iosrjce/papers /conf.15013/ Volume%2010/9.%2037-40.pdf?id=7557 , IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727 PP 37-40 www.iosrjournals.org.

[10]. Al-Janabi, Rasheed, M.A.-S., "Public-Key Cryptography Enabled Kerberos Authentication", IEEE, Develop- meashents in E-systems Engineering (DeSE), 2011.

[11]. Zvarevashe Kudakw, Mutandavari Mainford, Gotora Trust, , "A Survey of the Security Use Cases in Big Daa",http://www.ijircce.com/upload/2014/may/13_ASurv ey.pdf, International Journal of Innovative Research in Computer and Communication Engineering,(An ISO 3297: 2007 Certified Organization),Vol. 2, Issue 5, May 2014

[12]. Mehak, Gagandeep, "Improving Data Storage Security in Cloud using Hadoop", http://www.ijera.com/papers/ Vol4_issue9/Version%203/U4903133138.pdf, Int. Journal of Engineering Research and Applications, www.ijera.com ISSN: 2248-9622, Vol. 4, Issue 9(Version 3), September 2014, pp.133-138

[13]. Bhojwania Nikita, Prof. Vatsal Shahb," A Survey on HADOOP File System", http://ijiere.com/FinalPaper/ FinalPaper2014112822174540.pdf, International Journal of Innovative and Emerging Research in Engineering Vol- ume 1, Issue 1, 2014, e-ISSN: 2394 – 3343

[14]. Jorge Veiga, Roberto R. Exp´osito, Xo´an C. Pardo, Guillermo L. Taboada, and Juan Touri˜no. "Performance Evaluation of Big Data Frameworks for Large-Scale Data Analytics". In: 2015 IEEE International Conference on Big Data. IEEE Big Data'15. 2015, pp. 193–202.

[15]. Parmita Mehta, Sven Dorkenwald, Dongfang Zhao, Tomer Kaftan, Alvin Cheung, Magdalena Balazinska, Ariel Rokem, Andrew J. Connolly, Jacob VanderPlas, and Yusra AlSayyad. "Comparative Evaluation of Big-Data Systems on Scientific Image Analytics Workloads". In: Proceedings of the 43rd International Conference on Very Large Data Bases 10.11 (2017), pp. 1226–1237.

[16]. Aaron N. Richter, Taghi M. Khoshgoftaar, Sara Landset, and Tawfiq Hasanin. "A Multi-dimensional Comparison of Toolkits for Machine Learning with Big Data". In: 2015 IEEE International Conference on Information Reuse and Integration. IRI'15. 2015, pp. 1–8.

[17]. Jack Galilee and Ying Zhou. A study on implementing iterative algorithms using big data frameworks.url:https://webarchive.org/web/2016

0602063912/ http://sydney.edu.au/engineering/it/research/conversazione-2014/Galilee-Jack.pdf (visited on 08/19/2017).

[18].Shvachko K, Kuang H, Radia S, et al. The hadoop distributed file systemC// Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium

Cite this article as :

Nuzhat Shabir, Manoj Kumar Srivastava, "Security and Challenges in Implementation of Hadoop Technology in Cloud", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 4, pp. 252-261, July-August 2021.
Journal URL : https://ijsrcseit.com/CSEIT1836131