Big Data Platform Privacy and Security, A Review

Dana Shahin[1], Hannen Ennab[2], Reham Saeed[3], Jaber Alwidian[4]

[1,2,3,4]Computer Science, Princess Sumaya University for technology, Jordan

Abstract

big data represent information characterized by high volume, velocity and variety. It has been widely used to enhance decision making process due to insights can be extracted from big data. Implementing platform for big data on public cloud raise issues related to security and privacy. In this paper we investigate these issues from three perspectives: Data Privacy, Data Integrity and infrastructure security. We highlighted the main problems in each aspect and the proposed solution with its performance evaluation for implemented solutions.

Key words:

Data Mining, Cloud Storage, Integrity, MapReduce, Malicious worker

1. Introduction

In the last decades, the types and numbers of data were limited but now the amount of information is being increased [11]. The term big data has appeared to satisfy the big changes in information technology, big data is a huge mixture of data that you can extract meaning full information from. It is also known as immense and complicated data collection. Regarding this changes in data nature, normal database can't deal with big data as these data collections have extensive volume contain variant types of data which are generated by different sources at different rates [10].

Big data is one or more of the following types: either structured (the elements are arranged into a structure and they can be accessed easily, elements in the same group have similarities and unique description), unstructured (the elements are of any type, there are no organization rules) or semi-structured (this type lies between the previously mentioned types, data is not sorted out in conspicuous structure but it may have balanced data comprised of records) [11].

According to technology improvements, all organizations almost participate in data generation and many other organizations are concerned in data analysis to get benefit from it, also social media Revolution ease this process. In the other hand information is equal to money. This leads to loss of control over data flying everywhere. Data, data, data is all around.

As a result, big data privacy and security became very critical issues. Privacy refers to the privilege to own some management over how private data is gathered and how it is analyzed. It's the capability of one to prevent personal information from being known to other individuals. Security focuses mainly on how to protect data from attacks or abuse [2].

Table 1 shows the basic differences between privacy and security [2].

Table 1: Privacy vs security

Privacy	Security
Permitted/good use of individual's data/information	Availability, Integrity and Confidentiality
It usually refers to customers right to preserve their information from other unauthorized parties	It refers to confidentiality of enterprise/organizations
Good privacy leads to have a good security	It's possible to have a good security while having a bad privacy technology

Data privacy, Data integrity and Infrastructure security will be discussed in section 2, 3 and 4 respectively.

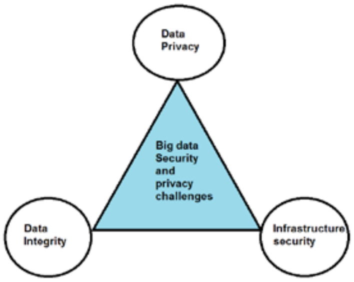

Fig. 1 Big data security and privacy challenges

2. Data Privacy

Data privacy should be maintained in order to set a limit for organization use without affect the main purpose of using big data systems.

As new technologies are being developed, social media data and data generated from sensors in many applications are increasing in unpredictable way. These large amounts of data are in different formats (structured, unstructured and semi- structured) as they are generated from different sources .sources difference also indicates that data is generated in variable rates i.e. sensors data is obtained in higher rates than google search entries. This lead data

Manuscript received May 5, 2019
Manuscript revised May 20, 2019

scientists to state big data V's term [1], those V's come from the properties of big data as explained.

V's of big data [1] [2]

1. Velocity: it is the speed of data generation (Streaming Data).
2. Variety: it refers to the difference in data types. Most of the current data are unstructured (images, voice, videos…etc.) rather than structured (phone, name, ID…etc.)
3. Volume: as explained above, huge amount of data are being generated. So the need for high storage and processing capabilities show up.
4. Value: if big data is captured and get analyzed well then it is converted into actionable insights this can produce a significant value which can help organizations to improve theirs decision. People who are responsible of extracting usable data from a larger set of any raw data are called **data minors.**

In typical **data mining** scenario there are four user roles [3]:

1. Data Provider: data owner.
2. Data Collector: is the user who brings the data from the providers then deliver it to data miner.
3. Data Miner: is the user who analyzes the data to get meaning full insights.
4. Decision Maker: the user who makes decisions based on data mining outputs.

To get benefit from big data, it must step into three phases which are called **big data life cycle**

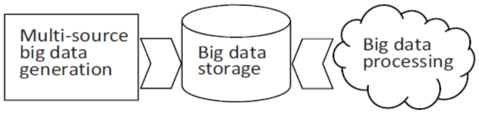

Fig. 2 Big Data life cycle

A. Data Generation

Traditional data is generated from specific sources like questioners or books, it is dedicated for specific purposes and it is structured. Big Data can be generated from different sources, since the generated data is large, diverse and complex it's hard to handle them with traditional systems. Data generation can be done either actively or passively. Active is like when the provider provides the data to a third party, submit a survey created by the data collector or even fill a specific form when creating a website account. While Passive data generation happens when data collector catch data generated by data provider's usual activities (i.e. browsing), data provider/owner may not be aware. Personal data are usualy gathered for business purpose i.e. online shopping can predict user's habits and a lot on personal information such as budget/salary [1].

1) Access Restriction

The major challenge in data generation phase is that how can the data provider protects sensitive data from undesired access In some cases data generator/provider needs to protect sensitive data in addition to participating in data mining. The data provider may want to provide the critical data to guaranteed data collector who prevents any unauthorized third-party from accessing the data. So if the provider are aware of how much benefit he can get when sharing data, he can decide the eligibility of providing sensitive data. As explained earlier, data comes from different sources (video, texts or images) so Data collector is also responsible of protecting collected data before it is being transformed/processed [3].

1. Browser's extension for anti-tracking: Users' online activity can be a good source for valuable information, so this is a perfect entry for internet companies who have a strong motivation to track people interactions on the internet. User can use extensions to stop trackers from using the cookies [3]. The main technology used for anti-tracking is: Do Not Track (DNT) [4], this allows users to prevent unvisited websites from tracking them. An HTTP header field called DNT is used for this property, if DNT is 1, this indicates that the user doesn't want the website to track him/her in 2009 an add-on to support DNT header in Firefox was created before many browsers supported DNT. (Do Not Track Me) and (Ghostery) [12] are examples of anti-tracking extensions [1] [3].
2. Script and Advertisement blockers extensions: These types of extensions kill scripts that send user's data to third parties and prevent site's Advertisement from appearing. Examples are AdBlock plus and NoScript [3].
3. Encryption tools: online communication can be hacked by third parties so users can use tools to encrypt messages and emails across the internet like (MailCloak9) and (TorChat). Also VPN (virtual private network) can be used for internet traffic encryption [3].

These tools limit access to private data but there is no guarantee that no untrustworthy side can access personal data so it's better to use anti-virus and anti-malware and to clear online traces always [1].

2) Data Distorting

As discussed above, internet users can't completely protect their data from unwanted access so instead of making a big effort trying to preserve the data, data can be distorted so the meaningful information can't be easily retrieved [1]. The below tools are used for data distorting:

1. Socketpuppet tool: this is used to hide user's true interactions and activities through internet, it falsifies user's identity. If multiple socketpuppets are used, the produced data/activity by single user will be considered as data belongs to different users. Finally the user's sensitive information such as political preference can't be discovered [5] [3].
2. Mask Tools: Mask Tools are used to mask user's identity and private information. For example when someone signs up for a website or wants to e-shop he/she has to enter information like phone number, email and financial data [3]. Tools like (MaskMe) help users to create aliases for personal information. Users have the option to use these aliases when information is required, so the websites don't get the actual information [6].

3) Data collector (Privacy preserving data publishing)

The main idea of this approach is that the data is considered as a private table with multiple records each has many attributes which have one of the four following types [7] [8]:

1. Attributes which are used to identify data's owner such as (ID, Name or mobile number).
2. Semi-Identifier attributes which are used with some external attributes to identify data's owner as shown in figure 3.
3. Attributes which are considered private from owner's perspective.
4. Normal attributes which are not 1, 2 or 3.

Before data is published to data minors/transformers, attributes of type 1 are removed and those of type 2 are modified (anonymized) so sensitive data will not be identified [1].

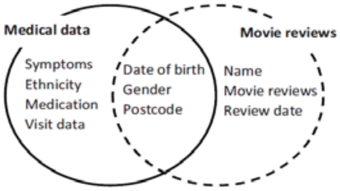

Fig. 3 Semi identifier attributes [1]

B. Data Storage

Small data is handled by traditional database, data computations like insertion and querying are done only through a specific interface integrated with the hardware storage. In big data systems data is stored in a distributed storage and there is a different query/processing data engine. So data storage for big data is divided into two parts: Hardware and Data management deployed on top of hardware infrastructure [13] [1].

The challenge is not about where to store data according to the enhancement in storage technology but it's about how to secure these data and protect it from threats. Since distributed environments deal with different data from wide range of data centers, the privacy issue is critical. According to the V's of big data, the scalability is important in big data storage infrastructure (i.e. it should be compatible with applications diversity), this lead us to virtualize the storage [1].

Storage virtualization means that there are multiple storage devices across the network as they are combined to appear as one storage device. This was achieved by the cloud service but the problem is that sensitive data will be managed by a third parties like cloud service provider, so the concentration will be on cloud data privacy [1] [3].

Issues of data in clouds come from [14]:

1. Many independent cloud users share the same physical infrastructure, this increase the probability that the attackers are on the same machine of the targeted data.
2. Sometimes data, data users and the application are all located in the cloud and the data owner does not have any control over his/her data, this allows to cloud provider to use the data for data mining purposes. In addition to that, there are no guarantees that when users delete their private data from the cloud the backed up copies in different data centers are deleted too.

In general there are four approaches to preserve data [9] as shown in table 2:

Table 2: Data preservation levels

File level	**Database level**	**Media Level**	**Application level**
It's applied on the host	It's applied on the data in database	It's applied on storage tools like hard disk	It's applied on the application
File encryption	Column encryption	Static data encryption	Ensures that only specific people can access the data through an application. (End to End) encryption

Sometimes inefficient because the sensitive data is mostly stored in small portion of the file.	It's cheap for organizations which store sensitive data in dedicated number of columns but there may be many encrypted data having the same value which make encryption process challenging.	The protection is limited since the encryption happens when data reaches the destination not through transmission, so it only limits the access to physical storage tool	It is very expensive.

To be more specific, the approaches below are used to maintain cloud data privacy:

1) Authentication

It's by default a mandatory technique, the most common way is using a username and a password by user to access the cloud service. When cloud provider checks the provided username and password it gives the user a lease to access the cloud. To access the very critical data, specific people are given a secret key after they got authenticated, in contrast to access public data you only need to be authenticated [18].

2) Encryption Based on Attributes

It is kind of end to end encryption, data owner sets a number of policies and data are encrypted under those policies so users who have attributes which are compatible with those policies can decrypt the data [1].
The most challenging point in this approach is policy updating, it requires data owner to share new policies with users, and data owner needs to retrieve data sent to cloud, encrypt it again with new policies and resend it again, this increase transmission overhead across the network, Approaches in [15] and [16] don't take into consideration policy updating but in [17] the author proposed an approach which the data owner request a privacy update from cloud without the need for cloud to decrypt the data and without moving the data back to the owner's local machine.

3) Encryption for storage

As [9] presents, the data is divided into portions series as each portion is stored on a separate storage media each has different cloud provider. If data owner wants to access these data, the divided parts are gathered to restore its original form.
This approach classifies the data on cloud into public or secret data. As the names indicate, public data does not need extra privacy plans thus all participant can access it without restrictions. In the other hand, secret data is not accessible by irrelevant organizations/people.

The proposed approach in [9] does not encrypt the whole data; instead it encrypts the storage path (cryptographic virtual mapping) of the data. Some applications require encrypting some parts of the data –which is considered very private- in addition to the storage path. Data owner always keeps information about storage indexes [1].

4) Computations Encryption

In this type, computational functions are computer over already encrypted data in the cloud and there is no need to decrypt the data in order to obtain encrypted results, outputs or predictions this approach is expensive to implement but guarantees high level of data privacy [19].

C. Data Processing

Some data scientists divide this stage into: collection, transmission, preprocessing and processing (get benefit of data) but we prefer to combine collection phase with data generation as discussed above.
Data transmission is the process of moving different types/numbers of data into a suitable storage. Preprocessing means data cleaning (to remove unnecessary and duplicated data). Processing is to transform and model the data in order to obtain meaningful information [1].
In this section we'll talk about privacy aspects while processing data (extract information /model).

1) Privacy in Data Clustering

The main idea of clustering is to divide unlabeled data into groups using set of features [3]. Traditional clustering technique assumes data to be in the same format, as it is processed by a single unit and this is not applicable to big data which is huge sized and is of variant types [1].[20] proposed a solution to cluster big data using multiple machines with map-reduce and parallel techniques.
One of the approaches used to preserve privacy in data clustering is derangement approach which modifies the data before applying clustering. Data modification is one of three types: scaling, rotation or translation. Data modification does not affect general features which are used for clustering. Translating is the process of adding a fixed –positive-value as a noise to each sensitive attribute. Scaling is to add a fixed-positive or negative- value as a noise for each sensitive attribute. Rotation deals with angles and the noise is defined by rotation angle. [21] has full detailed explanation about this approach.

2) Privacy in Data Classification

Classification aims to find the corresponding predefined label/category of the input data. It originally developed to work with traditional data in environments with a single processing unit (centralized). [22] Proposed a modification for the original classification algorithm to keep up with the needs of big data, this algorithm classifies the input data or it move it toward a second classifier. This is perfectly efficient when dealing with big data [1].

[23] Proposed an approach to ensure privacy. A random disturbance matrix is used to reconstruct the original data, thus the original data is hidden under altered data [1].

3. Data Integrity

Data integrity can be defined as the maintenance of data accuracy, consistency and trustworthiness during data life cycle, this is very important aspect as it will affect the decision making process.[24]

A. Data Collection

Data can be distorted in this stage due to many reasons including [25]

- Hardware/software tamper which will provide malicious input to central data collection system
- ID Cloning attack (Sybil attack)
- Providing malicious input by creating fake identities
- Input sources manipulation
- Transmission from input sources to central collection system

There are two major solutions [25]

- End-Point Input Validation

1. Trusted Platform Module (TPM)
   Used to guarantee integrity of raw sensor data but the problem is TPM no universally found in mobile devices and it can't handle threats related to input source manipulation.
2. Trusted certificate/trusted devices to prevent Sybil attack but managing certificates in a large enterprise setting with millions of entities is challenging

- Input Filtering
  Detect and filter malicious input using statistical methods because malicious input appear as outlier

B. Data Storage

Data on storage may corrupted due to many reasons like faults in storage device, network faults, or buggy software

- Internal integrity: So in Hadoop checksum is computed when data written to the disk for the first time and again checked while reading data from the disk. If checksum matches the original checksum then it is said that data is not corrupted otherwise it is said to be corrupted. [26]
- External integrity verification for outsourced big data in cloud and IoT.[27]

C. Data Processing

MapReduce is a programming model used to perform parallel processing on massive amount of data in open environments such as desktop grids, cloud computing and volunteer computing. Open environment suffer from problems related to privacy and security since the users only submitted their tasks and can't ensure the integrity of data being processed by workers in public cloud because the infrastructure no longer belongs to them. Also the long running processes increase the probability of data integrity corruption by attackers.

MapReduce framework consist of two major components master node and worker nodes. Worker nodes fall into one of two categories Mappers that generate intermediate results and reducers that generate final results. Mappers and reducers are susceptible to different attacks ,consequently worker nodes will generate incorrect results and may cause significant damage. This makes MapReduce integrity assurance challenge an essential issue.

Integrity assurance technique for different computing environments fall in one of three categories: Replication, Sampling and verification.

MapReduce integrity assurance pass through several stages through years 2009-2014:

1) Replication Based Techniques

In 2009, Wei et al. [28] proposed design and implementation for decentralized replication based integrity assurance framework, SecureMR. In this solution tasks are replicated between workers, attack is detected if there is inconsistency between results generated by different workers executing the same task. Design of this framework was introduced from two aspects architecture and communication. SecureMR architecture consist of five security components that provide set of security mechanisms. Communications between SecureMR components controlled using Commitment protocol and verification protocol. In commitment protocol mappers send commitment to master node. In verification protocol commitment sent to the master used to verify that intermediate results are consistent with these commitments with reducers help. This means that verification responsibility distributed among workers instead of being carried out only by master node. Several experiments has been done and detection rate was 90% with 40% of duplication rate.

when the bad worker fraction below 0.15 and cheat probability 0.5, on the other hand detection rate was 25% with 40% duplication rate when the bad worker fraction 0.5 and cheat probability 0.1, the maximum detection rate achieved under this environment is 80% with a duplication rate more than 500%. The main problem in this system that collusive workers can’t be detected because they cooperate to hide their attack. This was developed under the assumption that master and reducer are trusted nodes, this assumption might not be practical in real world, also it doesn’t perform well if most of workers are malicious.

In 2011 Mircea Moca et al [29] proposed design and implementation for distributed result checker which was mentioned in previous research [4] . In this study they employed the Majority Voting Method. This method detect malicious results by replicating the same task to multiple workersand the result returned by majority of workers considered correct. The master initiate verification for intermediate results generated by mappers and final results generated by reducers, The main issue in this implantation was not considering collusive workers and the trustworthiness of each worker. Ignoring worker trustworthiness will decrease performance less than 50% due to unnecessary computations.

In April 2012, Bendahmane et al [30][31] [32] proposed a solution to ensure map reduce integrity in open cloud computing environment. This approach use task replication and weighted t-first voting. The basic idea that results generated by different workers executing the same task are grouped based on result value then the result take from the first group with weight above threshold. Group weight calculated using equation (1) in [9], which consider the worker trustworthiness that depends on its computing behavior. The workers weight updated using equation (2) n [32]. In this study a dynamic blacklisting policy provided. Worker considered malicious and hence blacklisted if its error index using equation (3) in [32] exceed maximum error index. This ensure the accuracy and reliability of the mechanism. There proposed solution not evaluated. Both collusive and non-collusive attacks can be detected using this approach. This solution has no assumptions about workers trustworthiness as [1,2]. This solution can be enhanced by choosing a proper weight threshold, and deciding the maximum error index which minimize error rate and computation time. We noticed that same authors publish 3 different papers [30, 31,32] based on the same idea, and there was no new contribution

2) Replication and Verification

In 2011, Wang et al. [33] Replication and verification werecombined in Verification-based Integrity Assurance Framework (VIAF) for MapReduce to detect collusive and non-collusive workers.

Each task duplicated to two mappers in order to detect non-collusive mappers, in addition limited number of trusted nodes called verifiers added to verify consistent results, the credit of each mapper accumulated by passing verification. Mapper become trustable when its credit achieve quiz threshold. And mapper considered malicious once it fails any quiz.

VIAF was implemented on Apache Hadoop map reduce and achieve high computation accuracy (99.42% - 100%) for different quiz thresholds (1-7) , instead of 87.2% without verification. The verification overhead was acceptable for different quiz thresholds (19.83% - 22%), The main issue with this solution is that is assume reducers are trusted.

In 2011, Xiao et al.[34] proposed Accountable map reduce in cloud computing, an accountable map reduce employs an auditor group to perform accountability test (A-test) to detect malicious workers in real time. Auditor group acquire input data block and replay task on this data without knowing the working machine. Auditor consider a worker node as malicious if its output different from audit group output. Performance improved by using P-Accountability instead of 100% accountability as this required lower overhead.

In July 2013, Wang et al [35] proposed Cross Cloud MapReduce (CCMR), This framework consist of trusted master node runs on private cloud and normal worker nodes runs on public cloud. Master node verify data integrity on both phases map and reduce by using replication, verification and credit accumulation. The overhead problem resolved in this framework by minimizing cross-cloud communication. Accuracy was improved compared to secureMR which doesn’t perform well when most of worker nodes are malicious.

CCMR achieve 99.52% when malicious worker represent 16.7% of all nodes, and it add overhead 33.6%, according to these result overhead needs improvement.

In 2013 October Wang et al [36] proposed design, implementation and evaluation of IntegrityMR: an integrity assurance framework for big data analytics and management applications especially ApachePig. Integrity guaranteed at application layer and task layer. Task layer Experiments achieve 98% accuracy, with 5 as credit threshold, and overhead range from (18% to 82%). Application layer experiments on the other hand shows better performance (less than 35% of extra running time compared with the original MapReduce). The main issue with this solution is Distributed File System (DFS) bottleneck in cross-cloud environment

In 2014 Wang et al [37] Improve their work on VIAF which uses task replication and verification to ensure data integrity. In New implementation they didn’t assume reducers are trusted. The system perform well even the majority of workers are malicious. They also evaluate system performance using different variations related to environment, input size and application type. Also they expand their theoretical analysis.

In 2016, Wang er al [38] introduced MapReduce Computation Integrity with Merkle Tree-based verification (MtMR).This framework consist of master and verifiers run on private trusted cloud and normal workers run on public cloud. It ensure data integrity by applying merkle tree-based verification technique on map, reduce phases. It was able to detect semi-honest workers. Experiments showed that this architecture can assure high integrity with accuracy ratio 99.99% and moderate overhead because only 4% of records need to be processed on private cloud

3) Trusted Computing

Replication based approaches suffer from several issues like: large overhead , Probability-based fault discovery, Incapable of faulty-nodes identification,Vulnerable to user-based DoS attack, Trusted computing mechanism was introduced to overcome these problems. This approach will enforce workers to behave consistently using hardware that loaded with unique encryption key inaccessible to other nodes and special software [40].

In June 2012, Anbang et al[39] proposed design and implementation of trusted map reduce (TMR) framework to integrate MapReduce systems with TCG Trusted Computing infrastructure. In this framework integrity guaranteed by using remote certification. Latency reduced and scalability limitations eliminated by using a split and parallel verification schema. Proposed solution was implemented on the Hadoop MapReduce system. Experiments showed that a high strength integrity assurance has been achieved, and the overheads can easily be managed to less than 1% for an industry-strength implementation.

4) Watermarking-based approaches

In May 2012 , Chu Huang et al [41], proposed map reduce verification scheme for detecting cheating behavior of MapReduce computing in the context of text processing problems. Approach was built based on watermark injection and weighted sampling methods, there is no constraints that master and/or reducer have to be trusted.

Table 3: Mapreduce Integrity

Year	Technique	Mechanism	Assumptions	Implementation	Performance	Contribution	Limitations
2009	SecureMR	Task replicated to multiple workers and attack detected if there is inconsistency between results returned for same task executed by different workers	DFS provide integrity protection Master is trusted and has public key Reducer is trusted Mappers not trusted Good workers always return correct results while bad workers behave arbitrarily Each worker has public/private key	Non-blocking verification scheme Hadoop MapReduce 11 workers , 1 master	90% Detection rate, 40% Duplication Rate, 1.5 cheat probaility < 15% malicious workers 25% Detection rate, 40% Duplication rate, 0.5 cheat probanility 50% malicious workers 0%Detection rate, Any duplication rate, 100% bad workers	Reduce duplication rate based on probability models	Collusive workers can’t be detected Assuming Master,Reducer are trusted may be impractical in real world Doesn’t perform well when majority of workers are malicious
2011	Distributed Result Checker using Majority Voting Mechanism (MVM)	Task replicated to multiple workers and the result returned by majority of workers considered a correct result	Master is trusted Non-Collusive untrusted mappers/reducers System run on desktop grid infrastructure	Not implemented	Not stated	Use majority voting mechanism	Collusive workers can’t be Detected Worker trustworthiness not considered

2012	Task Replication and weighted t-first weight voting	Task replicated to multiple workers and grouped based on result value then correct result taken from group that exceed threshold	DFS, master are trusted Mappers, Reducers not trusted Communication network is trusted	Not implemented	Not stated	Using weighted t-first voting mechanism based on equations	Malicious workers no more than half of all workers

Year	Technique	Mechanism	Assumptions	Implementation	Performance	Contribution	Limitations
2011	VIAF	Task replicated to two mappers to detect non-collusive mappers Verifier added to verify consistent results of collusive mappers, mapper trustworthiness increased each time its pass verification and considered trusted if it exceed certain threshold	DFS, master, verifiers, reducers are trusted Mappers not trusted	Hadoop MapReduce 1 master 1 verifier 4 collusive workers 5 good workers Word count application, 400 map tasks, 1 reduce task	99.42% - 100% Accuracy Compared to 87.2% without verification 19.83% - 22% verification overhead	Add Verification	Assume Reducers are trusted
2011	Accountable Map Reduce	Auditor group employed to verify worker accountability by executing tasks and comparing the results with workers results	Auditor group is trustworthy domain Workers are malicious	Not implemented	Not stated	Auditor group verification	
2013	CCMR	Replication, Verification and Credit accumulation	DFS, Master, verifiers are trusted Workers are malicious	Implemented on Apache Hadoop MapReduce	99.52% accuracy 33.6% overhead 16.% malicious workers	Master and verifiers on private cloud while workers on public cloud	Moderate overhead
2013	IntegrityMR	MapReduce on top of hybrid clouds which consists of one trusted private cloud and multiple public clouds	DFS, Master are trusted Workers are malicious Communication network trusted	Apache Pig Apache Map Reduce	Task layer :98% accuracy (18% - 82%) overhead Application layer: < 35% extra running time compared to map reduce		
2012	TMR	Trusted computing using hardware and software	-----	Not implemented	Not stated	Overhead reduced because there is no replication	Hard to implement on public cloud sue to lack of flexibility

4. Infrastructure Security

Now we will discuss the security during the infrastructure of the system and when coming to cover this topic we must start the discussion with the frameworks that found to secure the architecture of a big data system , and since Hadoop is the most commonly used we will start with it , then we will highlight other topics such as communication security and architecture security.

A. Security in Hadoop

Hadoop servers trust anyone that can reach them on the network and intruders can monitor network traffic , so there will be a security risks that have to be solved by :

1) Authentication

It is the central of any security effort and it is one of the most critical aspects in every security issues because without it we cannot identify who should access the data so, authentication is a must in big data platforms such as

Hadoop, in a simple logic, it means verifying a username and password. Hadoop does not have built in capabilities to authenticate the user, it integrates with some other tools and reuse them to achieve users and process authenticity such as:

a. Kerberos

Is a protocol used for authentication over the network. It uses secret-key cryptography to offer durable authentication for client/server applications[42], so Hadoop's user can prove its identity.
Kerberos consists of the below components:

1. key distribution center (KDC) which consists logically of three parts:
- Authentication server: authenticates the user and issues a ticket Granting.
- Ticket granting server: the application server of KDC which provides service ticket.
- Database: stores principals and other data. Having a valid TGT means that Authentication server verified your credential, at the end before access the Hadoop cluster you have to get a service ticket from TGS.
2. Clients which include users, hosts and services.
3. Server which consists of service providers requested to start session.

b. LDAP (Lightweight Directory Access Protocol)

Is a protocol used in internet programs to allow them to look up data from a server, LDAP is a directory-like information that means specific database implement for frequent queries but it does not work with infrequent updates.
There are two choices for implemented **LDAP**:

- simple authentication .
- security layer.

LDAP consists of : Bindings, policy engine and policy provider.

2) Authorization

Many people believe that authorization is similar to authentication, but it is a much different than authentication, authorization means that what user can or cannot do within a Hadoop once being authenticated [44], while as mentioned earlier authentication is for identifying who should access the data Authorization has been achieved in Hadoop in different ways such as:

a. Sentry Apache

Is a Hadoop authorization engine which provides the authenticated users the permission to control the data accessing. [45] [46] Components of authentication process using Sentry Apache divided into :

- Sentry Server
- Data Engine
- Sentry Plugin

b. Apache Ranger

Apache Ranger can offer a entire technique to protection for a Hadoop cluster. It offers a centralized platform to outline, administer and control safety policies continuously throughout Hadoop components. [46]
Components of Apache Ranger :

- Ranger admin portal
- Ranger plugins
- User group sync

Table 4: Sentry vs ranger

Apache Criteria	Sentry	Ranger
Owner	Cloudera	Hortonworks
Support impala	Support	Not support
Support Hdfs, Solr ,Hive	Support all of them	Support all of them
support column-level permissions in Hive	Lower granularity for columns or cells	Yes it includes column-level permissions in Hive.

So based on your requirement you decide which apache is suitable to use , the main point is what Hadoop distribution tool that you are using like Cloudera or Hortonworks. And are you need column level security or not, However both of Apache Senrty and Apache Ranger very close to each other

1) Data Protection

Which means how to protect data that have been stored in Hadoop cluster and keep them protected when transferring it, data protection possibly will achieve by encryption.
The following methods are used for **encryption**:

a. Novel method

To encrypt document while being transferred. In this strategy, firstly data which is to be transferred to HDFS is put away in a buffer, then encryption is connected to the buffer's data before being sending it to HDFS. The data in the file will be in the byte format after encryption and in order to decrypt the content you have to factorize that large numbers into four unique prime numbers .This encryption is straightforward to user.[47][48]

b. Fully Homomorphic encryption

Technique underpins the administration of cipher textual content facts beneath the security insurance,
Furthermore, can legitimately connected to ciphertexts for recovery and calculation in the clouds [49][50]. it's far a

form of encryption with a further evaluation capability for calculating over encrypted data without get right of entry to the secret key.[51][52]

Finally we can summary that, there are three areas to Implement security in Hadoop:

Table 5: Security areas in hadoop

Security areas in Hadoop	What is it	Tools
Authentication	Guarantees only real user, provider accesses cluster.	Kerberos, LDAP and so on.
Authorization	Make sure what user and application can do with data.	Apache Sentry, Apache Ranger, etc.
Data Protection (Encryption)	Protect data from unauthorized access .	Homomorphic encryption technology , etc.

B. Communication Security

It one of the most important aspect have to be considered when speaking about security in big data world , however there are a few papers describe it deeply and too many papers ignore it .It related to communication between the parts of big data system.

Kerberos and Secure Socket Layer (SSL) are some of available solutions for obtaining communication security among different nodes. [53]

Table 6: KERBEROS vs SSL

Kerberos	SSL
Private key encryption is used.	Public key encryption is used.
Works based on the trusted third party.	Works based on certificate.
Open source and free.	The service is not free.
Key revocation can be accomplished by disabling a user at the authentication server.	Key revocation requires revocation server to keep track of bad certificate.

C. Architecture Security

It is a different aspect we have to be interested in it to achieve highly security in big data. Architecture includes data models, data management, data storage and data analysis tool [54] which it means architecture security related to security in all parts of architecture mentioned above . To design new platform or changing one of the current architecture we have to understand data life cycle for users and classify important components and tasks, so we can easily summary interactivity of security concerns .We will figure out one of the current big data vendors and how it achieved the security in it's own architecture which is IBM.

IBM security products and service such as

IBM Identity Governance and Intelligence

Within your organization, you would like to be able to perceive WHO has access to what and the way that access is being employed. Is your identity governance operating intelligently?IBM isconcentrated on assembling and analysing identity knowledge to support enterprise IT and restrictive compliance. With IGI, you'll improve visibility into however access is being utilized, grade compliance actions with risk-based insights, and build higher choices with clear unjust intelligence. All of this is often driven by a business-activity primarily based approach to risk modeling, a significant person for IBM that produces life easier for auditors and risk compliance managers.

IBM zSecure Audit

IBM® Security zSecure™ Audit measures and verifies the effectiveness of mainframe security policies for IBM Resource Access management Facility (RACF®), CA-ACF2 and CA prime Secret Security. zSecure Audit generates reports to quickly find issues related to a selected resource — like AN unprotected knowledge set — to produce vulnerability analysis of your mainframe infrastructure. It additionally provides a compliance framework for testing against business laws. As a result, you'll be able to cut back errors and improve overall quality of services.

5. Conclusion

The volume and transmission speed of data has been increased in recent years and the need for new systems to store and process it has been shown, thus we have to pay attention to all data related problems that may appear like security and privacy issues. In this paper we have covered many possible of them such as data privacy, data integrity and data infrastructure security. First we discussed the data privacy in data generation, data storage and data processing phases, then we discussed the data integrity in the same previous phases, and we finalized our discussion with infrastructure security including security in Hadoop, communication security and architecture security. At the end we are looking forward to reach the level of having a fully secured big data solutions starting from a-to-z.

References

[1] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo, "Protection of big data privacy," IEEE Access, vol. 4, no. c, pp. 1821–1834, 2016.

[2] P. Jain, M. Gyanchandani, and N. Khare, "Big data privacy: a technological perspective and review," J. Big Data, vol. 3, no. 1, 2016.

[3] L. Xu, C. Jiang, J. Wang, J. Yuan, and Y. Ren, "Information security in big data: Privacy and data mining," IEEE Access, vol. 2, no. January, pp. 1151–1178, 2014.

[4] O. Tene and J. Polonetsky, "To Track or 'Do Not Track': Advancing Transparency and Individual Control in Online Behavioral Advertising," Ssrn, pp. 281–357, 2011.

[5] S. Kumar, J. Cheng, J. Leskovec, and V. S. Subrahmanian, “An Army of Me: Sockpuppets in Online Discussion Communities,” 2017.

[6] “MaskMe,” Google. [Online]. Available: https://chrome.google.com/webstore/detail/maskme/dpkiidbpeijnaaacjlfnijncdlkicejg. [Accessed: 28-Apr-2019].

[7] R. Liu and H. Wang, “Privacy-preserving data publishing,” Proc. - Int. Conf. Data Eng., vol. 42, no. 4, pp. 305–308, 2010.

[8] R. C.-W. Wong and A. W.-C. Fu, “Privacy-Preserving Data Publishing: An Overview,” Synthesis Lectures on Data Management, vol. 2, no. 1, pp. 1–138, 2010.

[9] H. Cheng, C. Rong, K. Hwang, W. Wang, and Y. Li, “Secure big data storage and sharing scheme for cloud tenants,” China Commun., vol. 12, no. 6, pp. 106–115, 2015.

[10] D. Viji, K. Saravanan, and D. Hemavathi, “A journey on privacy protection strategies in big data,” Proc. 2017 Int. Conf. Intell. Comput. Control Syst. ICICCS 2017, vol. 2018–Janua, pp. 1344–1347, 2018.

[11] R. Pandya, V. Sawant, N. Mendjoge, and M. D ’silva 4, “Big Data Vs Traditional Data,” vol. 3, no. X, pp. 192–196, 2015.

[12] “Ghostery – Privacy Ad Blocker,” Google. [Online]. Available: https://chrome.google.com/webstore/detail/ghostery-—-privacy-ad-blo/mlomiejdfkolichcflejclcbmpeaniij?hl=en. [Accessed: 29-Apr-2019].

[13] H. Hu, Y. Wen, T. S. Chua, and X. Li, “Toward scalable systems for big data analytics: A technology tutorial,” IEEE Access , vol. 2, pp. 652–687, Jul. 2014.

[14] A. Gholami and E. Laure, “Security and Privacy of Sensitive Data in Cloud Computing : A Survey of Recent Developments,” pp. 131–150, 2015.

[15] K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie, “DAC-MACS: Effective data access control for multiauthority cloud storage systems,” IEEE Transactions on Information Forensics and Security, 20-Dec-2016. [Online]. Available: https://scholars.cityu.edu.hk/en/publications/dacmacs(fcd6e5f4-486e-42dd-8aba-bbf8b529f7b2).html. [Accessed: 29-Apr-2019].

[16] K. Yang and X. Jia, “Expressive, efficient, and revocable data access control for multi-authority cloud storage,” IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 7, pp. 1735–1744, 2014.

[17] K. Yang, X. Jia, and K. Ren, “Secure and Verifiable Policy Update Outsourcing for Big Data Access Control in the Cloud,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 12, pp. 3461–3470, 2015.

[18] A. Kumar, B. G. Lee, H. Lee, and A. Kumari, “Secure storage and access of data in cloud computing,” Int. Conf. ICT Converg., no. October, pp. 336–339, 2012.

[19] C. Gentry, “A fully homomorphic encryption scheme,” Proc. 41st Annu. ACM Symp. Symp. theory Comput. - STOC ’09, no. September, p. 169, 2009.

[20] A. S. Shirkhorshidi, S. Aghabozorgi, T. Y. Wah, and T. Herawan, “Computational Science and Its Applications – ICCSA 2013,” vol. 7975, no. June, 2013.

[21] R. R. Rajalaxmi and A. M. Natarajan, “An effective data transformation approach for privacy preserving clustering,” J. Comput. Sci., vol. 4, no. 4, pp. 320–326, 2008.

[22] C. Tekin and M. Van Der Schaar, “Distributed online big data classification using context information,” 2013 51st Annu. Allert. Conf. Commun. Control. Comput. Allert. 2013, pp. 1435–1442, 2013.

[23] S. Agrawal, J. R. Haritsa, and B. A. Prakash, “FRAPP: A framework for high-accuracy privacy-preserving mining,” Data Min. Knowl. Discov., vol. 18, no. 1, pp. 101–139, 2009.

[24] “Data integrity,” Wikipedia, 08-Feb-2019. [Online]. Available: https://en.wikipedia.org/wiki/Data_integrity. [Accessed: 27-Apr-2019].

[25] Cloud Security Alliance, “Expanded Top Ten Big Data Security and Privacy Challenges,” Cloud Secur. Alliance, no. April, pp. 1–39, 2013.

[26] “Data Integrity and Availability in Apache Hadoop HDFS,” Hortonworks, 12-Jun-2018. [Online]. Available: https://hortonworks.com/blog/data-integrity-and-availability-in-apache-hadoop-hdfs/. [Accessed: 01-May-2019].

[27] C. Liu, C. Yang, X. Zhang, and J. Chen, “External integrity verification for outsourced big data in cloud and IoT: A big picture,” Futur. Gener. Comput. Syst., vol. 49, pp. 58–67, 2015.

[28] W. Wei, J. Du, T. Yu, X. Gu, N. Carolina, and U. States, “SecureMR : A Service Integrity Assurance Framework for MapReduce,” 2009 Annu. Comput. Secur. Appl. Conf., pp. 73–82, 2009.

[29] M. Moca, G. C. Silaghi, and G. Fedak, “Distributed results checking for mapreduce in Volunteer Computing,” IEEE Int. Symp. Parallel Distrib. Process. Work. Phd Forum, pp. 1847–1854, 2011.

[30] A. Bendahmane, M. Essaaidi, A. El Moussaoui, and A. Younes, “Computaion integrity mechanism for MapReduce in cloud computing system,” Proc. 2nd Natl. Days Netw. Secur. Syst. JNS2 2012, pp. 74–79, 2012.

[31] A. Bendahmane, M. Essaaidi, A. El Moussaoui, and A. Younes, “A new mechanism to ensure integrity for MapReduce in cloud computing,” Proc. 2012 Int. Conf. Multimed. Comput. Syst. ICMCS 2012, pp. 785–790, 2012.

[32] A. Bendahmane, M. Essaaidi, A. El Moussaoui, and A. Younes, “Result verification mechanism for MapReduce computation integrity in cloud computing,” Proc. 2012 Int. Conf. Complex Syst. ICCS 2012, pp. 1–6, 2012.

[33] Y. Wang and J. Wei, “VIAF: Verification-based integrity assurance framework for MapReduce,” Proc. - 2011 IEEE 4th Int. Conf. Cloud Comput. CLOUD 2011, pp. 300–307, 2011.

[34] Z. Xiao and Y. Xiao, “Accountable MapReduce in cloud computing,” 2011 IEEE Conf. Comput. Commun. Work. INFOCOM WKSHPS 2011, pp. 1082–1087, 2011.

[35] Y. Wang, J. Wei, and M. Srivatsa, “Result Integrity Check for MapReduce Computation on Hybrid Clouds,” 2013 IEEE Sixth Int. Conf. Cloud Comput., pp. 847–854, 2014.

[36] Y. Wang, J. Wei, M. Srivatsa, Y. Duan, and W. Du, “IntegrityMR : Result Integrity Assurance Check Framework for Big Data Analytics and Management Applications How to Ensure High Result Integrity for Pig ? • How do we construct big data analytics infrastructure on the cloud that can provide high integrity ,” p. 2013, 2013.

[37] Y. Wang, J. Wei, and Y. Duan, “Securing MapReduce Result Integrity via Verification-based Integrity Assurance

Framework," Int. J. Grid Distrib. Comput., vol. 7, no. 6, pp. 53–70, 2015.

[38] K. Ghosh, "Big Data: Security Issues, Challenges and Future Scope," Int. J. Res. Stud. Comput. Sci. Eng., vol. 3, no. 3, pp. 1–11, 2016.

[39] A. Ruan and A. Martin, "TMR: Towards a trusted MapReduce infrastructure," Proc. - 2012 IEEE 8th World Congr. Serv. Serv. 2012, pp. 141–148, 2012.

[40] C. A. A. Bissiriou and M. Zbakh, "Towards Secure Tag-MapReduce Framework in Cloud," Proc. - 2nd IEEE Int. Conf. Big Data Secur. Cloud, IEEE BigDataSecurity 2016, 2nd IEEE Int. Conf. High Perform. Smart Comput. IEEE HPSC 2016 IEEE Int. Conf. Intell. Data S, pp. 96–104, 2016.

[41] C. Huang, S. Zhu, and D. Wu, "Towards trusted services: Result verification schemes for MapReduce," Proc. - 12th IEEE/ACM Int. Symp. Clust. Cloud Grid Comput. CCGrid 2012, pp. 41–48, 2012.

[42] P. Savitra, J. Padwal, J. Chaitali, M. Surabhi Nilangekar, and U. K. Bodke J, "Automated Attendance System in College Using Face Recognition and NFC," Int. J. Comput. Sci. Mob. Comput., vol. 6, no. 6, pp. 14–21, 2017.

[43] B. Saraladevi, N. Pazhaniraja, P. V. Paul, M. S. S. Basha, and P. Dhavachelvan, "Big data and Hadoop-A study in security perspective," Procedia Comput. Sci., vol. 50, pp. 596–601, 2015.

[44] N. Sirisha and K. V.D. Kiran, "Authorization of Data In Hadoop Using Apache Sentry," Int. J. Eng. Technol., vol. 7, no. 3.6, p. 234, 2018.

[45] Sentry.apache.org. (2019). Apache Sentry. [online] Available at: https://sentry.apache.org/ [Accessed 7 May 2019].

[46] M. Gupta, F. Patwa, J. Benson, and R. Sandhu, "Multi-Layer Authorization Framework for a Representative Hadoop Ecosystem Deployment," no. March 2018, pp. 183–190, 2017.

[47] O. O'Malley, K. Zhang, and S. Radia, "Hadoop security design," Yahoo, Inc., Tech. …, no. October, pp. 1–19, 2009.

[48] G. P. Patro, "A Novel Approach for Data Encryption in Hadoop A Novel Approach for Data Encryption in Hadoop."

[49] White, T. (2012). Hadoop. Beijing [etc.]: O'Reilly Media.

[50] En.wikipedia.org. (2019). Homomorphic encryption. [online] Available at: https://en.wikipedia.org/wiki/Homomorphic_encryption [Accessed 7 May 2019].

[51] Wang, G., Atiquzzaman, M., Yan, Z. and Choo, K. (2017). Security, Privacy, and Anonymity in Computation, Communication, and Storage. Cham: Springer International Publishing.

[52] R. R. Parmar, S. Roy, D. Bhattacharyya, S. K. Bandyopadhyay, and T. H. Kim, "Large-Scale Encryption in the Hadoop Environment: Challenges and Solutions," IEEE Access, vol. 5, no. October 2018, pp. 7156–7163, 2017.

[53] M. Onuralp Gökalp, K. Kayabay, M. Zaki, A. Koçyiğit, P. Erhan Eren, and A. Neely, "Big-Data Analytics Architecture for Businesses: a comprehensive review on new open-source big-data tools," no. October, 2017.