Acta Scientific COMPUTER SCIENCES

Volume 5 Issue 1 January 2023

Research Article

State-of-the-art in Cloud Computing Data Security

Saugat Singh and Gajendra Sharma*

School of Engineering, Department of Computer Science and Engineering, Kathmandu University, Dhulikhel, Kavre, Nepal

***Corresponding Author:** Gajendra Sharma, School of Engineering, Department of Computer Science and Engineering, Kathmandu University, Dhulikhel, Kavre, Nepal.

Received: October 13, 2022

Published: December 26, 2022

Abstract

Increase in cloud users raise different issues in cloud computing. Among them one of the crucial issues is security. Due to the increase in users of cloud computing in several areas, the size of data stored in the cloud also increases drastically. Data Security in the cloud is raising challenges for Cloud Service Providers (CSP) and Cloud Service Consumer (CSC). There is a lot of research happening to make data secure in the cloud but researchers are still not able to set a standard requirement for data security in the cloud because of technological advancement and the use of the cloud in different fields and purposes. This paper is based on a review of the different state-of-the-art in cloud computing data security. Further, those approaches are analysed with security requirements for data security in cloud computing.

Keywords: Cloud Service Provider (CSP); Data Owner (DO); Service Level Agreement (SLA); Advanced Encryption Standard (AES)

Introduction

Cloud computing is one of those new-generation web technologies that is built on utility computing, virtualization, service-oriented architecture, parallel and distributed computing for resources, and knowledge sharing. In cloud computing, there are mainly three parties, namely Cloud Service Provider (CSP), Data Owner (DO), and Users (Hogan, M., and Sokol, A., 2013) According to the National Institute of Standards and Technology [1], "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort and or service provider interaction".

Cloud computing is divided into two parts such as the deployment model and the service delivery model [2]. The development model of cloud computing is Private, Public, Community and Hybrid clouds. Similarly, the service delivery model of cloud computing is Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Cloud computing has five special characteristics namely Rapid Elasticity, Measured Service, Resource Pooling, Board Network access and On-Demand Self Service. Besides these features one of the biggest challenges in cloud computing is security. There are several issues in the security of cloud computing among them data security is the most important one. To maintain data security in cloud computing following security challenges must be addressed efficiently.

- **Security challenges in CIA Triad:** Data is the key component of any organisation, CIA losses may have a significant impact on the cloud computing sector.
- **Security challenges in the Authentication and Access Control (AAC):** The process of verifying and confirming a user's identity in order to connect, access, and use cloud resources is known as authentication and access control (AAC).
- **Security challenges Due to Broken Authentication, Session and Access Controls:** The threat of broken authentication and session control arises from the application domain's incorrect implementation of authentication and session management.
- **Other Data-Related Security Issues**: Other minor data security issues can arise in cloud computing due to data location, multi-tenancy, and backup.

Ensuring data security in cloud based social networks [3]

Authors convince us to shift social networks to the cloud such as user demand of low latency data access, data consistency, availability and privacy but network service providers have one big limitation of capital which directs social network providers to use cloud services. While moving to the cloud numerous issues come in front namely data issues, infected applications, privacy issues and security issues etc. Among the security issues are the most challenging ones. So, the author investigated and proposed a framework for secure sharing of data in social networks using cloud services.

A general idea of this framework is that the sender encrypts data and sends it to the proxy server, again encrypted data is re-encrypted before sending it to the cloud and if the receiver wants to decrypt the data then the receiver should have a valid id. Valid id means an encrypted and signature authenticated certificate issued by a certificate authority to both sender and receiver user. This scheme employed Recoverable Hierarchical Identity-Based Encryption (RHIBE) and it has seven different algorithms as System setup, Encryption, Initial key extract, Initial key delegate, Time key update, Time key delegate and Decryption. Similarly, AES is adopted for the encryption process and proxy re-encryption is used to encrypt already computed ciphertext.

Initial work on this framework starts from CSP providing services to the users based on SLA. After using cloud service in social networks, the network provider is responsible to secure their user's data efficiently now the proposed framework comes into action. In this scheme, the General Key Manager (GKM) is responsible to compute the keys and distribute them among users and the proxy. GKM generates a public and private key for encryption and decryption by employing an Identity-based Algorithm. Similarly, GKM authenticates user ID and user encrypt the data with public key and Proxy stands between user and cloud and re-encrypt the ciphertext with the symmetric key generated through AES algorithm by GKM. GKM is timely updated about users joining and detaching from the group on this basis user status GKM shares the key.

On the other hand, if the user wants to access data then the user should authenticate with GKM after authentication GKM will provide a public key to authenticated users. After user verification proxy will retrieve data from the cloud using the keyboard search technique and pre-decrypt the data using a symmetric key. Then if users want to read the data then they can decrypt the data using a private key.

The algorithm for this framework is

- Alice encrypts the plaintext T using the Public key (see), i.e. C(T)=ENC(T).
- The proxy re-encryption C(T) with the symmetric key(sk) i.e.Cp(T)=REENC(C(T)).
- Send the data to the cloud after re-encryption.
- Pre-decryption of the stored data by the proxy using a symmetric key (sk), i.e., C(T)=PREDEC(Cp(T)).
- Authentication of the receiver User ID by GKM.
- Decryption by the user through his private key(rdk) i.e. T=DEC(C(T)).

The proxy cannot read the data and only can re-encrypt and pre-decrypts the data with the symmetric key. The proxy can only convert in one direction thus this scheme is unidirectional. Collusion is completely avoided as GKM shares a private key with the user who wants to access data.

Improving database security in cloud computing by fragmentation of data [4]

The paper raises a database confidentiality issue in cloud computing. To solve that issue, the concept is taken from the nature of cloud computing, which is a decentralised system in integration with the encryption algorithm. In this paper, the authors employed a combination of encryption algorithms and fragment techniques. This approach has two different clouds: Private cloud and Public cloud. Private cloud has users and proxy servers while public cloud consists of one master cloud and many slave clouds. Master cloud is responsible to stores an encrypted replication of the entire database and each slave or public cloud is responsible to store extended columns. This scheme used five different encryption algorithms based upon the ability to support users queries. The encryption algorithms used are Advanced Encryption Standard (AES), Order-Preserving Encryption (OPE), Homomorphic Encryption (HOE), Search Encryption and Deterministic Encryption.

Here fragmentation is done on master cloud and slave cloud. The steps they use for fragmentation are:

- The entire database is encrypted using a suitable secure encryption algorithm and doesn't disclose the encryption key to the master cloud provider.
- After the relation of the master cloud is generated an additional column or attribute beside the primary key is created with a unique index to the tuples of the relation.

- Slave cloud is configured with vertical fragmentation to create a variable number of replicas of the columns are stored in slave clouds. Columns stored in the slave cloud are encrypted but do not disclose the encryption keys to the cloud provider.

The master cloud is a key factor that maintains entire relations using the AES-CBC encryption algorithm. Here only one column, an index is stored in plaintext which acts as a candidate key to query the relation and fetch desired tuples. A proxy server is a vital component located inside a private cloud and performs almost all processing such as creation, insertion, decryption, query parsing and retrieval of results and communication between the private cloud with the outside is done through SSL. When a proxy gets a client query, it parses and transmits it as subqueries to slave clouds' expanded columns. Each slave cloud responds with a set of indices that make up the answer to the sub-query it received. If the query's where clause has more than one where condition, the proxy conducts a union or intersection algorithm on the indices when it receives the results of subqueries. Once the final results' indices have been created, the proxy sends a query to the master relation to retrieve the tuples that match them. Proxy also encrypts all inserted value and selection queries.

The proposed scheme is evaluated by analysing total delay using an analytical model and through modelling. This approach i.e. Secured distributed approach is compared with two base methods unsecured method and secured centralised method where data is stored in plain text and after encrypted respectively. The proposed approach is experimented with the above two approaches in terms of delay due to communication, crypto processing (encryption/decryption), query processing and total delay. For this, they use analytical as well as emulation methods. For the experiment they took three different sets of queries: Select statement in which Where clause contains one simple condition, two conditions and three conditions connected with AND Boolean operator.

The experiments are applied for the Unsecured Centralised, Secured Centralised and Secured Distributed Methods. In Query1, the Secure centralised method has much fewer delays because of information in clear text and no decryption. In Query2, the effect of communication is increased much more as the number of predicate increase to two and the number of cloud increase to three (one master and two slaves). In Query3, the number of predicates increased to three and the number of clouds increased to four (one master three slaves) which clearly depicts the most increase in communication delay.

For result analysis, they compare the delay of each component of the analytical model with the corresponding delays of the emulation modelling across all queries for every approach.

- **Findings 1 (Communication Delay):** Communication delays between the analytical and emulation model is higher in query 3 than in queries 1 and 2.
- **Findings 2 (Query processing Delay):** Small selectivity delays through analytical modelling are higher than those obtained through emulation. For higher selectivity delays obtained by emulation are much higher than those obtained by analytical modelling.
- **Findings 3 (Crypto Delay):** Crypto delays are proportionally affected by the message size. For this approach, decryption is done only at the final result retrieved from the master cloud so that the size of the decryption work depends on the selectivity factor. Thus, delays obtained by the analytical model and emulation model are the same for both the Secure centralised and Secure distributed approaches.
- **Findings 4 (Proxy Delay):** The proxy delays include delays to perform the intersection of results returned from slave clouds and overhead associated in transforming the user query to queries on clouds and also processing results received from the clouds. Proxy delays in both models (analytical and emulation) are much less or fewer than the delays of the other components.

There are several types of delays on the internet so delays in cloud infrastructure are also variable because they are dependent on the demand, the resources supplied by the cloud provider, and the cloud provider's quality of software that measures the load and balances the allocation of resources to handle it. As a result, while analytical modelling may not accurately forecast actual delays when compared to actual delays observed in emulation, it does provide suggestions on the overall trend when variables such as selectivity or the type of sent queries are altered.

An improved attribute-based encryption technique towards the data security in cloud computing [5]

The paper tries to address the resources and knowledge sharing issue for the cloud computing environment. The solutions for those issues are proposed as the use of attribute-based encryption (ABE), distributed hash table (DHT) network and identity-based time-release encryption (IDTRE). System model and security requirements are considered important for the proposed scheme. For this proposed scheme, the system model has categorised seven entities namely CSP, DO, User, Attackers, Trusted third party, Time server and Node of the DHT network. Here, nodes of the DHT network are responsible for managing and storing key shares. Similarly, security requirements are Data confidentiality, Collision resistance, Resistance to attacks, Sensible data should be accessed before the desired release time and sensitive data should be deleted after the expiration time.

The purpose of the scheme is to restrict hopping attack, brute-force attack, sniffing attack and traditional cryptanalysis attack. They analyse the security of this scheme on five security requirements terms:

- **Security analysis for Data confidentiality:** Here each authorised user is assigned a unique key ‘r’ so attackers or CSP cannot perform the decryption process unless they satisfy access policy and authorised users can only access a sensitive key in authorised period and data are self-destructed after a predefined time.
- **Security analysis of collision resistance:** By using a key generation algorithm, the trusted parties generate different r values for different users.
- **Security analysis of the DHT network:** This approach is secure against Sybil attacks because Sybil attacks occur by the Hopping attacks and Sniffing attacks. But this scheme addresses the hopping attacks and sniffing attacks by increasing the length of the ciphertext using the IDTimeReEncrypt algorithm by encrypting the decryption key.
- **The Security analysis of the encryption algorithm:** In this proposed scheme the access tree is exposed in the ciphertext and use the AttributeScrm algorithm to re-encrypt the ciphertext and gets another ciphertext after that the IDTimeReEncrypt algorithm encrypt the decryption key. So, this scheme is more secure compared to Identity Based Self Destructive Scheme (ISS) and SSDD.
- **The security analysis of the system model:** Encryption of encryption exists in this proposed model and the AttributeScrm algorithm conceal the attributes values of the access tree because the ciphertext includes the access tree in the plaintext form. So, this model is more secure from various attacks.

The proposed model in comparison with ISS, Safe Vanish, Vanish and SSDD on eight different security properties shows that the proposed model is far better than others. Similarly, the performance of the proposed model is compared with the ISS and SSDD on the basis of computational overhead, encryption and decryption time for different numbers of attributes and encryption and decryption time for different file sizes shows that the proposed scheme has better performance than the other two.

Towards DNA based data security in Cloud Computing environment [6]

(Namasudra, Devi, Kadry, Sundarasekar, and Shanthini, 2020) proposed Deoxyribonucleic Acid (DNA) based scheme which is a biological concept in data security for the cloud computing environment. There are four phases to accomplish data security procedure in this approach namely:

- **System setup:** The service provider chooses a big prime number to identify the multiplicative group. From this multiplicative group CSP selects public and private keys for both DO and users. Public keys are publicly available while authorised DO and authorised entities know data owner public key and CSP public key respectively. However, an individual entity’s private key is kept secret from other entities.
- **User Registration and Login Phase:** For registration in the cloud, users must send their personal information then CSP receives the request and collects shared information and generates a user’s profile. Then SSL is established between the cloud and users to deliver key pairs and registration replies. The CSP relies on acknowledgement to the user after a login process then only big data accessing requests can be sent.
- **DNA based Big Data Storage Phase:** This is the backbone of the purposed model which have two phases under this:
    - **DNA based secret key generation:** At first, the user communicates with DO and a session is generated then DO verifies the user authorization if a user is valid then DO fetches all information such as user_id, MAC address etc and based on that data key is generated. DO converts those values in the 8-bit binary form by using decimal encoding rule then the 8-bit binary value of resultant divides into two parts and perform XOR operation between these two parts. Each 8-bit binary number are transformed into their corresponding ASCII values and ASCII values are

again transformed to actual 8-bit binary values. The result is divided into two parts and add the first part to the left side of the MAC address while a second part is added to the right side of the MAC address. Then, DO convert the result into binary number and if it is not 1024-bit then extra bits are deleted from the left side in case of exceeding or 0 is added to the right side in case of less bit. Then, those 1024-bit binary numbers are partitioned into four equals 256-bit blocks and assigned DNA bases of each part and apply complementary rules on it which is the final DNA based secret key (DNABSK). DNA bases refer to A (Adenine), C (Cytosine)), G (Guanine) and T (Thymine) on which data are encrypted and stored.

- **DNA Based Big Data Encryption:** After splitting plain text into 1024-bit blocks into four groups with each block of 256-bit. XOR operation is executed between 256-bit encrypted plain text and a 256-bit key. Then each 2-bit of a binary number is converted into the DNA base, as there are 4 DNA bases (A, C, G, T) so there may be a maximum of 24 combinations and DO can choose any combination out of 24 which transforms the encrypted binary form of plain text into the DNA bases. Then, DO apply the complementary rule on the resultant binary bases. Finally, the DO again encrypts the resultant DNA sequence by using the DO private key and CSP public key. Also DO encrypts the corresponding data access right or certificate along with big data and saves the entire encrypted file or data on the cloud database.
- **Big Data Access Phase:** The user must send a registration request to the CSP to access big data from the cloud server then CSP sends the registration reply after that the user can login into the cloud server to access big data.

The authors insist this model is secured against main attacks and security analysis of DNA based data security is checked on five different attacks.

- **Malware Injection:** DO randomly generates a DNA secret key and uses this secret key for encrypting big data again the complementary rule is applied for key generation and big data encryption and all credentials are shared only with the authorised user in encrypted form. So, attackers cannot decrypt the data without credentials thus unable to access big data by malware injection.
- **Side-channel attack:** The respective user is only able to get the secret key when they decrypt by using the user private key and DO public key. If any hacker places or replaces any VM, data security still cannot be compromised because DO does not store DNA secret keys in the cloud server. Thus, the proposed method is secure against this attack.
- **Phishing attack:** When the user sends an access request for big data, the service provider only gives the respective DO public key to the user in ciphertext form. The DO only provides the DNA secret key after confirming the user's authenticity. Both the DO and CSP do not disclose any sensitive or confidential data of themselves to others and communication between CSP and users happen through SSL. Hence, this approach can resist phishing attacks.
- **Insider attack:** This scheme is secure against insider attacks because DO creates the DNA secret key based on users attribute and MAC addresses and employs complementary rules in the big data encryption process. So, it is very challenging for hackers or authorised users to get all secret information.
- **Denial of Service attack:** In the proposed scheme, the DO randomly generates a 1024-bit DNASK and encrypts big data by using this long 1024-bit key. The DO also uses DOPrK and CSPPuK to encrypt the big data before storing it on the cloud server. If the attackers attack the cloud server by DoS, they cannot get original data content because the DO shares all the credentials and complementary rules in the encrypted form only with the authorised users after checking the user's authenticity. Thus, users' confidential or sensitive data does not face any security issue and the proposed scheme can be considered secure against DoS attacks.

In order to check the performance of the proposed scheme, 50 experiments are executed between purposed model (DNABDS), reversible data hiding scheme (RDHS), zigzag Morse code based ACM (ZMCACM) and client-side data encryption scheme (CSDES) based on a number of users vs key generation time, key retrieval time, data encryption time and data decryption time. Dataset is taken from CityPulse Datasets collection. The proposed scheme takes a little bit more time to generate a secret key for new users and for old users key generation time is reduced however other existing schemes have linear curves for key generation. Similarly, for the key retrieval time of the proposed scheme existing users have the corresponding data and they do not need to get the DNA secret key from the DO while new users have to get the DNA secret key from the DO thus the key retrieval time increases for first accessing. However, for other scheme user must execute many operations to retrieve the secret key to and users have to get a secret key from DO every time they wish to access big data.For encryption time, the proposed scheme DO uses a complementary rule to encrypt big data after generating a DNA secret key. In ZMCACM, Morse code is used to encrypt big data and encrypted data is stored in a zigzag line after many operations using 'dash' and 'dot'. Similarly, in RDHS classical algorithm of reversible data protecting or data hidden approach is used. For CSDES scheme uses many XOR operations to encrypt big data. Comparing all other schemes with the proposed scheme, the proposed scheme takes less big data encryption time. For big data decryption, the proposed scheme takes less time than another scheme because DO provides all the credentials to the users at the time of providing the secret key and users can easily decrypt big data using these credentials. There is no XOR operation in the proposed scheme for big data decryption but other schemes use many EXOR operations for big data decryption.

AuthPrivacyChain: A blockchain-based access control framework with privacy protection in the cloud [7]

Yang., *et al*. proposes a blockchain-based system of privacy control called AuthPrivacyChain. In this scheme first the account node address is used as the identifier in the blockchain when redefining the cloud access control authorization. The account node address is then encrypted and stored in the blockchain. The processes for access control, permission and revocation authorization are developed for AuthPrivacyChain. They are also adopting AuthPrivacyChain, which is based on the Enterprise Operating System (EOS). The findings reveal that AuthPrivacyChain can prevent unauthorised access to resources by hackers and administrators, as well as preserve approved privacy. Only users with access rights can utilise this scheme to gain access to services. As a result, this strategy satisfies confidentiality, openness, accessibility, honesty, and responsibility while also preventing external users and internal management from attack.

The proposed system models consist of four entities namely Cloud, Blockchain, Data owner (DO) and Data User (DU). The procedures used in this model are:

- **Initialization:** This step consists of three entity registrations:
    - **Registration of Cloud:** Cloud sends a request for registration to Blockchain. Blockchain then calls a respective function to generate an asymmetric key of wallet address (identity of cloud), create a wallet address and automatic synchronization. Then, the cloud calls function to save cloud info into Blockchain.
    - **Registration of User:** The registration of DO and DU is called user registration. First, a user sends a registration request to Blockchain. Blockchain then replies with an asymmetric key of wallet address (identity of users) and user address to users. The user will self-update the local database by synchronization.
    - **Resource Publishing:** DO calls resource upload permission to cloud. Cloud returns resource information to DO and decrypts it with a symmetric key and gets resource information. DO calls blockchain with a private key, address, cloud location, public key and resource information to publish the information of resource registration.
- **Access Control:** This step is responsible to validate the authentication of users on resources stored in the cloud. First, a user sends a request for resources they wish to access. Cloud then decrypts it and computes a hash function for a smart contract to get resource information in Blockchain and send it to Blockchain. Blockchain replies with encrypted resource information to the cloud. Cloud decrypt resource information and verify with resource information by the user if matched then the access is granted otherwise access is denied.
- **Authorization:** This term is classified into two parts:
    - **Direct Authorization:** If the owner of the resources grants access to other users that are called Direct authorization. DU1 sends an authorization request to cloud and cloud decrypts to get resource information, data owner address and address of user1. Cloud extract hash id and check whether an address of user1 has authorised permission to access user2. Blockchain returns resource information in Blockchain to the cloud. Cloud then verifies the DO address is indeed a host of resources and confirms that the permission requested by DU1 is within the scope of authorization if no process is terminated but if positive then Cloud sends an authorization request of DU1 to DO. DO publish authorization to Blockchain. Finally, DO sends the authorization flag to DU1.
    - **Indirect Authorization:** The approved DU authorises

other users indirectly, including at least five entities: DO (resource owner), DU1 (authorizer), DU2 (requester), Blockchain, and Cloud.

- **Authorization Revocation:** The revocation of authorization by authorised users is referred to as authorization revocation. DU1 sends the information of authorization revocation to Cloud. Cloud checks whether or not the address of user1 is authorised to address user2 in Blockchain. Blockchain returns encrypted resource information to the cloud. Cloud decrypts authorization to get resource information. If resource information is not null then the cloud sends authorization revocation to DU1. DU1 revokes authorization. DU1 then sends a notification to DU2 about permission revocation.

The security analysis is performed on Confidentiality, Integrity, Availability, Authenticity and Accountability of the proposed model. The information sent between the DU, DO, Cloud, and Blockchain is encrypted, and the access control permission is likewise encrypted and kept in the blockchain, so the AuthPrivacyChain's assure confidentiality and privacy. AuthPrivacyChain can offer users system integrity, ensuring that the system can fulfil the intended functions in a regular manner and avoiding unauthorised tampering, whether deliberate or inadvertent. AuthPrivacyChain can function quickly and cannot deny authorised users access. Because they must first publish their own certificate to join the system, DU, DO, Cloud, and Blockchain in AuthPrivacyChain can be validated and trusted (public key). AuthPrivacyChain is capable of keeping activity records of entities in Blockchain that allows it to perform post-audit analysis and keep track of security events.

Additionally, access control performance evaluation is evaluated using identity authentication, authorization, access permission, and audit. The result shows that the access AuthPrivacyChain and traditional access control have fairly comparable performance. Traditional access control is done in the cloud, whereas AuthPrivacyChain's access control requires blockchain interaction.

Findings

Each proposed state-of-the-art in cloud computing data security has a different approach to address the data security issues. The table below shows the different security challenges covered by the above state-of-the-art in cloud data security.

Conclusions

SN	Paper	Confidentiality (C),Integrity (I) and Accessibility (A)			Authentication (Auth)and Access Control (AC)		Broken Authentication, session and Access Control	Other Data-Related Security Issues
		C	I	A	Auth	AC		
1	Paper 1 (Praveena and Smy)	✓	✓	✓	✓	X	X	X
2	Paper 2 (Alsirhani, Bodorik, and Sampalli)	✓	✓	✓	X	X	X	X
3	Paper 3 (Namasudra, 2017)	✓	✓	✓	✓	✓	X	X
4	Paper 4 (Namasudra, Devi, Kadry, Sundarasekar, and Shanthini, 2020)	✓	✓	✓	✓	✓	X	X
5	Paper 5 (Yang., *et al.*)	✓	✓	✓	✓	✓	✓	X

Table 1: Security Analysis of each State-of-the-art .

Cloud computing is an emerging technology that provides on-demand service and pay-as-per-use policy. Heterogeneity, resource sharing, multi-tenancy, virtualization, mobile cloud computing, and SLA are examples of cloud environments that make cloud security more vulnerable [2]. The huge drawbacks of cloud computing are the security and privacy of data. Anything that is more precious in a digital world is data. So, no one wishes to dare their data because it could cost immeasurable harm. The main motive of this paper is to assess distinct approaches in cloud computing data security. In this paper, five state-of-the-art in cloud computing are reviewed and discover distinct approaches used for data security. Those approaches are further analysed with security challenges such as confidentiality, integrity, availability, authentication, access control and so on. At last, security requirements each state-of-the-art satisfies for data security in cloud computing are tabularized which depicts each proposed model work towards addressing different security issues.

Source Selection

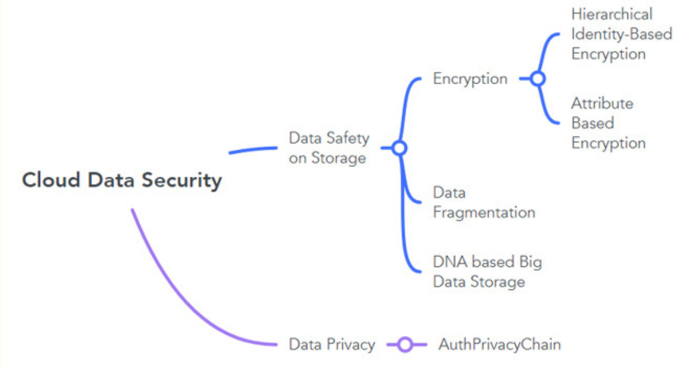

Figure 1: Mind Map.

Different online databases were queried with the search term of "Cloud Computing" + "Security" and the papers were selected based on the relevancy to the database security and the publication date tracing back to the recent decade. Results concerning the authentication and access of the database were selected for the research. Furthermore, the query results were considered state of the art based on the number of times the results were cited by other papers.

- Parveen http://dx.doi.org/10.1109/iceca.2017.8212819 30 citations
- Alsirhani http://dx.doi.org/10.1109/comapp.2017.8079737 14 citations
- Namasudra https://doi.org/10.1002/cpe.4364 56 citations
- Namasudra https://doi.org/10.1016/j.comcom.2019.12.041 81 citations
- Yang https://doi.org/10.1109/access.2020.2985762 58 citations

Limitations

Paper 1

- The architecture and the algorithm seem quite good but there is no performance analysis of this state of the art so it seems quite incomplete.
- This framework employs data encryption technology for data security but isn't concerned about Denial of Service (DoS) attacks as data encryption can't address DoS attacks.
- Did not cover attack resistant analysis.
- The overall security performance depends on the General Key Manager (GKM) but no alternative measures in case of GKM failure.

Paper 2

- This experiment is in a small scale of query just three and only three predicates are experimented thus can also get different results in the large and complicated query.
- This paper approaches vertical fragmentation only and ignores the possibilities of horizontal fragmentation.
- Attack resistant analysis was not performed.
- Authorization and authentication like security issues are not defined.

Paper 3

- Security issues on data backup are not addressed.
- Data Authentication policy was not employed.
- Security challenges on broken authentication, session and access control are not analysed.

Paper 4

- Although this scheme proposed clear analysis for data security in the cloud with experiments, mathematical proof of security analysis is not mentioned.
- Security issues like broken authentication, session, access control and backup data security are not taken into account.

Paper 5

- Attack resistant analysis was not mentioned.
- Blockchain consumes high energy power consumption analysis was not answered.
- Challenges that occur during integration between cloud and blockchain was not noted.

Bibliography

1. NIST cloud computing standards roadmap. National Institute of Standards and Technology (2013).

2. Kumar PR., *et al*. "Exploring data security issues and solutions in cloud computing". *Procedia Computer Science* 125 (2018): 691-697.

3. Praveena A and Smys S. "Ensuring data security in cloud based social networks". 2017 International Conference of Electronics, Communication and Aerospace Technology (ICECA) (2017).

4. Alsirhani A., *et al*. "Improving database security in cloud computing by fragmentation of data". 2017 International Conference on Computer and Applications (ICCA) (2017).

5. Namasudra S. "An improved attribute-based encryption technique towards the data security in cloud computing". *Concurrency and Computation: Practice and Experience* 31.3 (2017): e4364.

6. Namasudra S., *et al*. "Towards DNA based data security in the cloud computing environment". *Computer Communications* 151 (2020): 539-547.

7. Yang C., *et al*. "AuthPrivacyChain: A blockchain-based access control framework with privacy protection in cloud". *IEEE Access* 8 (2020): 70604-70615.