Robust and Auditable Secure Data Access Control in Clouds

KARPAGADEEPA.S[1], VIJAYAKUMAR.P[2]

Research Scholar, Department of Computer Science, Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science, Coimbatore-641005, India.

Research Supervisor, Head, Department of Computer Applications, Sri Jayendra Saraswathy Maha Vidyalaya College of Arts and Science, Coimbatore-641005, India

ABSTRACT: In distributed computing, accessible encryption strategy over Auditable data is a hot research field. Be that as it may, most existing system on encoded look and auditable over outsourced cloud information and disregard customized seek goal. Distributed storage space get to manage is imperative for the security of given information, where information security is executed just for the encoded content. It is a smaller amount secure in light of the fact that the Intruder has been endeavored to separate the scrambled records or Information. To determine this issue we have actualize (CBC) figure piece fastening. It is tied in with adding XOR each plaintext piece to the figure content square that was already delivered. We propose a novel heterogeneous structure to evaluate the issue of single-point execution bottleneck and give a more proficient access control plot with a reviewing component. In the interim, in our plan, a CA (Central Authority) is acquainted with create mystery keys for authenticity confirmed clients. Not at all like other multi specialist get to control plots, each of the experts in our plan deals with the entire trait set independently. Keywords: Cloud storage, Access control, Auditing, CBC.

Keywords:
Robust and Auditable Secure Data, Access Control, Clouds

1.INTRODUCTION

Distributed storage is a promising and critical administration worldview in distributed computing. Advantages of utilizing distributed storage incorporate more prominent openness, higher dependability, quick arrangement and more grounded assurance, to give some examples. Notwithstanding the said benefits, this worldview likewise delivers new difficulties on information get to control, which is a basic issue to guarantee information security. Since distributed storage is worked by cloud specialist organizations, who are typically outside the put stock in area of information proprietors, the conventional access control techniques in the Client/Server demonstrate are not appropriate in distributed storage condition. The information get to control in distributed storage condition has subsequently turned into a testing issue. To address the issue of information get to control in distributed storage, there have been many plans proposed, among which CBC is viewed as a standout amongst the most encouraging methods.

A notable element of CBC is that it gifts information proprietors coordinate control in view of access arrangements, to give adaptable, fine-grained and secure access control for distributed storage frameworks. In CBC conspires, the entrance control is accomplished by utilizing cryptography, where a proprietor's information is scrambled with an entrance Structure over traits, and a client's mystery key is marked with his/her own particular properties. Just if the traits related with the client's mystery key fulfill the entrance structure, can the client decode the comparing figure content to get the plaintext. Up until now, the CBC based access control plans for distributed storage have been created into two reciprocal classes, to be specific, single-expert situation, and multi specialist situation.

Albeit existing Encrypted get to control plans have a considerable measure of alluring highlights, they are neither strong nor proficient in key age. Since there is just a single expert accountable for all qualities in single-specialist plans, disconnected/crash of this expert makes all mystery key solicitations inaccessible amid that period. The comparable issue exists in multi-expert plans,

Manuscript received May 5, 2024
Manuscript revised May 20, 2024
https://doi.org/10.22937/IJCSNS.2024.24.5.11

since each of numerous specialists' deals with a disjoint quality set. In single-expert plans, the main specialist must check the authenticity of clients' characteristics previously producing mystery keys for them. As the entrance control framework is related with information security, and the main qualification a client have is his/her mystery key related with his/her characteristics, the procedure of key issuing must be careful. In any case, in reality, the traits are different. For instance, to check whether a client can drive may require an expert to give him/her a test to demonstrate that he/she can drive. Consequently he/she can get a property key related with driving capacity.

To manage the confirmation of different traits, the client might be required to be available to affirm them. Moreover, the procedure to confirm/dole out ascribes to clients is generally troublesome with the goal that it regularly utilizes heads to physically deal with the confirmation, as has specified, that the credibility of enrolled information must be accomplished by out-of-band (for the most part manual) implies. To settle on a cautious choice, the unavoidable cooperation of people influences the confirmation to tedious, that causes a solitary point bottleneck particularly, for a huge framework, there are constantly extensive quantities of clients

LITERATURE SURVEY:

(1) Improving Privacy and Security in Decentralized Cipher text-Policy Attribute-Based Encryption by Jinguang Han, Willy Susilo, Yi Mu, Jianying Zhou, Man Ho Allen Au.

Abstract:

In past security protecting multi-specialist property based encryption (PPMA-ABE) plans, a client can obtain mystery keys from numerous experts with them knowing his/her properties and moreover, a focal specialist is required. Prominently, a client's character data can be removed from his/her some touchy qualities. In our PPDCP-ABE contrive, each master can work independently with no planned push to starting the structure and issue puzzle keys to customers. Additionally, a customer can get riddle keys from different experts without them knowing anything about his overall identifier (GID) and qualities.

(2) Efficient Decentralized Attribute-based Access Control for Cloud Storage with User Revocation by Jianwei Chen, Huadong Ma.

Abstract:

Distributed storage get to control is vital for the security of outsourced information, where Attribute-based Encryption (ABE) is viewed as a standout amongst the most encouraging advancements. Ebb and flow looks into for the most part concentrate on decentralized ABE, a variation of multi-specialist ABE plot, in light of the fact that regular ABE plans rely upon a solitary expert to issue mystery keys for all of clients, which is exceptionally unreasonable in a huge scale cloud. Initially, our plan measurements not require any focal expert and worldwide coordination among various specialists. At that point, it bolsters any LSSS get to structure and therefore can encode information regarding any Boolean equation. Our security and execution examination show the exhibited plan's security quality and proficiency as far as adaptability and calculation.

(3) TMACS: A Robust and Verifiable Threshold Multi-Authority Access Control System in Public Cloud Storage by Wei Li, Kaiping Xue, Yingjie Xue, Jianan Hong.

Abstract:

Trait based Encryption (ABE) is viewed as a promising cryptographic leading apparatus to ensure information proprietors' immediate control over their information out in the unfasten cloud storage. The prior ABE plans include just a single expert to keep up the entire characteristic set, which can bring a solitary point bottleneck on both security and execution. Security and execution examination comes about demonstrate that TMACS isn't just obvious secure when not as much as t specialists are traded off, yet in addition vigorous when no not as much as the experts are alive in the framework. Besides, by proficiently joining the conventional multi-expert plan with TMACS, we build a half breed one, which fulfills the situation of properties originating from various specialists and accomplishing security and framework level strength.

(4) Secure and verifiable policy update outsourcing for big data access control in the cloud by Kan Yang, Xiaohua Jia , Kui Ren.

Abstract:

Because of the high volume and speed of enormous information, it is a viable alternative to store huge information in the cloud, as the cloud has capacities of putting away huge information and preparing high volume of client get to demands. Characteristic based encryption (ABE) is a promising procedure to guarantee the conclusion to-end security of huge information in the cloud. Notwithstanding, the arrangement refreshing has dependably been a testing issue when ABE is utilized to build get to control plans. In this paper, we propose a novel plan that empowering effective access control with dynamic arrangement refreshing for enormous information in the cloud. We focus on working up an outsourced methodology invigorating strategy for ABE structures. Our strategy can dodge the transmission of scrambled information and limit the calculation work of information proprietors, by making utilization of the beforehand encoded information with old access arrangements.

3.PROPOSED METHODOLOGY

It is a littler sum secure in light of the way that the Intruder has been tried to isolate the mixed records or Information. To decide this issue we have complete (CBC) figure piece securing. It is tied in with adding XOR each plaintext piece to the figure content square that was at that point conveyed. The result is then mixed using the figure estimation in the standard way. Each subsequent figure content piece depends upon the previous one.

The essential plaintext piece is added XOR to a subjective instatement vector. We propose a novel heterogeneous structure to clear the issue of single-point execution bottleneck and give a more capable access control plot with an exploring segment.

Our structure uses various credit specialists to share the store of customer genuineness check. Then, in our arrangement, a CA (Central Authority) is familiar with make riddle keys for realness affirmed customers. Not in the slightest degree like other multi authority get the chance to control plots, each of the specialists in our arrangement manages the whole attribute set autonomously.

To overhaul security, we in like manner propose an assessing segment to perceive which AA (Attribute Authority) has mistakenly or maliciously played out the legitimacy check framework. Examination shows that our structure guarantees the security necessities and also rolls out exceptional execution improvement on key age.

Security and execution examination comes to fruition exhibit that it isn't recently evident secure when not as much as specialists are exchanged off, yet furthermore overwhelming when no not as much as masters are alive in the structure. We show that our approach achieves bring down correspondence, figuring and limit overheads, appeared differently in relation to existing models and plans.

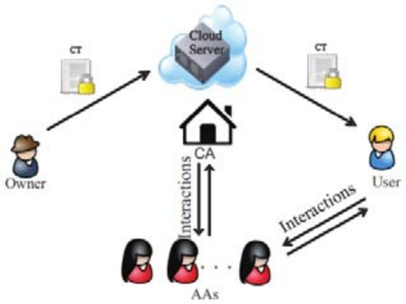

Figure 3.1: Architecture Diagram

3.1 MODULE DESCRIPTION

The framework model of our outline, which includes five elements:

- A central authority (CA)
- Attribute Authorities (AAs)
- Data Owners (Owners)
- Data Users (Users)
- Data Upload
- Data Download
- Auditing and Tracing

3.1.1 CENTRAL AUTHORITY

CA is the leader of the entire framework. It is in charge of the framework development by setting up the framework parameters and producing open key for each trait of the widespread property set. In the framework instatement stage, it doles out every client an extraordinary aid and each property specialist a one of a kind Aid. For a key demand from a client, CA is in charge of creating mystery keys for the client on the premise of the got middle of the road key related with the client's authentic properties confirmed by an AA. As a director of the whole framework, CA has the ability to follow which AA has inaccurately or perniciously checked a client and has allowed ill-conceived characteristic sets.

3.1.2 ATTRIBUTE AUTHORITIES

AAs are in charge of performing client authenticity check and creating middle of the road keys for authenticity confirmed clients. Dissimilar to a large portion of the current multi-specialist plans where every AA deals with a disjoint characteristic set individually, our proposed conspire includes various experts to share the obligation of client authenticity confirmation and every AA can play out this procedure for any client autonomously. At the point when an AA is chosen, it will check the clients' true blue properties by difficult work or confirmation conventions, and produce a moderate key related with the qualities that it has authenticity checked. Middle key is another idea to help CA to create keys.

3.1.3 THE INFORMATION PROPRIETOR (OWNER)

Information Owner characterizes the entrance strategy about who can access each record, and encodes the document under the characterized emerge. As an issue of first significance, each proprietor encodes his/her data with symmetric encryption estimation. At that point, the proprietor figures get to arrangement over a quality set and scramble the symmetric key under the approach as indicated by open keys acquired from CA.

3.1.4 THE INFORMATION SHOPPER (USER)

The information client is appointed a worldwide client character Uid by CA. The client has an arrangement of characteristics and is furnished with a secrecy key related with his/her trait set. The customer can uninhibitedly get any fascinated encoded data from the cloud server.

3.4.5 DATA UPLOAD

The confirmed Data Owner has been transferring the Data utilizing Chain Block Cipher strategy. The chain piece figure has been scrambled with the circling procedure. The cloud server doesn't direct information get to control for proprietors. The encoded information put away in the cloud server.

3.1.6 DATA DOWNLOAD

The Verified Data client has been downloading with their mystery keys. The data will be retrieved from the server.

3.1.7 AUDITING AND TRACING

Every AA may produce a middle of the road key for any quality set related with a particular client, and after that CA can create the mystery key for this client with no more confirmation. Be that as it may, AAs can be traded off and can't be completely trusted. In the mean time, the client authenticity check is directed by physical work, and subsequently AAs may perniciously or inaccurately produce a middle key for an unconfirmed trait set. A malignant client will attempt any conceivable intends to pick up the mystery key related with the particular credit set to acquire the information get to authorization. Under this supposition, the client would regularly indicate unusual practices. More often than not, we have to hold the responsibility of AAs to keep the bargained or got rowdy ones from openly creating mystery keys for pernicious clients.

3.2 SECURITY ASSUMPTIONS AND REQUIREMENTS

In our proposed plot, the security suppositions of the five parts are given as takes after. The cloud server is constantly on the web and oversaw by the cloud supplier. More often than not, the cloud server and its supplier are thought to be "straightforward yet inquisitive", which implies that they will accurately execute the undertakings allocated to them for benefits, however they would endeavor to discover however much mystery data as could reasonably be expected in view of information proprietors' sources of info and transferred records.

CA is the head of the whole framework, which is constantly on the web and can be thought to be completely trusted. It won't connive with any element to secure information substance. AAs are in charge of directing authenticity confirmation of clients and judging whether the clients have the guaranteed properties. We accept that AA can be bargained and can't be completely trusted. Moreover, since the client authenticity check is led by physical work, mis-operation caused via indiscretion may likewise happen. In this way, we require an examining system to follow an AA's bad conduct. In spite of the fact that a client can openly get any scrambled information from the cloud server, he/she can't decode it unless the client has traits fulfilling the entrance arrangement inserted inside the information. Subsequently, a few clients might be deceptive and inquisitive, and may intrigue with each other to increase unapproved access or attempt to plot with (or even trade off) any AA to acquire the entrance authorization past their benefits.

Proprietors approach control over their transferred information, which are secured by particular access arrangements they characterized. To ensure secure access control out in the open distributed storage, we assert that an entrance control plot needs to meet the accompanying four fundamental security prerequisites:

- **Data secrecy**: Information content must be kept secret to unapproved clients and additionally the inquisitive cloud server.
- **Collusion-protection**: Malevolent clients intriguing with each other would not have the capacity to consolidate their credits to unscramble a cipher text which each of them can't decode alone.
- **AA responsibility**: An inspecting component must be formulated to guarantee that an AA's misconduct can be recognized to keep AAs' manhandling their energy without being identified.
- **No ultra viruses for any AA**: An AA ought not to have unapproved energy to straightforwardly produce mystery keys for clients. This security necessity is recently presented in view of our proposed various leveled system.

4. EXPERIMENTAL RESULTS

The thesis has used C#.Net for developing the front end of this software and SQL Server for the back end. The reason for using C#.Net is its flexibility. This can add or remove any features without editing the whole code. This separated the standalone functions like port matching and IP address matching in separate functions which are reused again and again. For the back end this needed a freely distributed and powerful database so SQL Server was a good choice. Whole of the games will be stored in the database.

In this Article, presented Robust Auditable access with cloud based on a single authority algorithm, where the authority is replaced by one CA and multiple AAs (we rename the combination as CA/AAs unit). Practically, our methods can also be built in a lot of traditional multi-authority settings, e.g., where the attribute authorities (referred to as TAAs to distinguish AAs of traditional multi authority settings from AAs in CA/AAs unit of our system) manage disjoint attribute subsets.

4.1 COMPARISION OF EXISTING AND PROPOSED SYSTEM

As we have specified, truly, the repetitive system of client authenticity confirmation is considerably more muddled than mystery key age. In our plan, the heap of authenticity confirmation is shared among different AAs, while a substantially lighter computational undertaking is appointed to the single CA. In this manner, the proficiency of key dispersion is made strides. All the more Specifically, numerous AAs are standby for the authenticity check in the framework. At the point when there is a key demand, a sit out of gear AA is chosen by a planning calculation to play out the check and different AAs are standby to serve the consequent client demands.

We give the hypothetical execution examination as the accompanying advances. Right off the bat, we show our framework in queueing hypothesis, and afterward we break down the state probabilities to acquire the two critical variables, the mean disappointment likelihood and the normal sitting tight time for clients.

4.1.1 MODELING IN QUEUING THEORY

For simplicity, we assume there is a central coordinator which assigns users' key requests to *AA*s. The coordinator maintains each *AA*'s state with the boolean value of 0/1, where state 0 indicates that the *AA* is available to conduct verification, and state 1 indicates the *AA* is occupied and is not available right now. Each time the coordinator assigns a key request to an *AA* with the state 0. If all *AA*s are busy, the new users who are requesting the secret keys will wait in a queue to be served. The coordinator can adopt *First Come First*

Service (FCFS) algorithm to serve the arriving users. It's important to note that some other strategies can also be adopted in our architecture, such as a user arriving at a nearest *AA* according to his/her knowledge and decision. Thus, each *AA* may separately maintain a queue of its own. However, this model may not achieve load balance as some *AA*s may be unoccupied while other *AA*s are always busy in serving users' requests. Therefore, we introduce a central coordinator and adopt a single arrival queue as our strategy. The queueing model of our system is shown in Fig. 3, and can be treated as a Markov process. The central coordinator is deployed at the entrance of the system to monitor each *AA*'s state (occupied/unoccupied) and assign each arriving users to an unoccupied *AA*. Furthermore, we model our system as follows. On *AA*s' side, the queueing model can be described as *M/M/C/N/*∞, where *C* is the number of *AA*s, *N* is the capacity of our system and *N* =*C* + *K* (*K* is the queue length that indicates the maximum number of the queued users.). Here, the first *M* describes that arrivals of key requests follow a Poisson process in the system, and the second *M* means the verification service times are exponentially distributed. ∞ means the source of key requests is infinite. When there are *N* users in the system, other new arrivals of users' requests will be rejected. This property can ensure that a user will not wait in the queue for an irrationally long time. On *CA*'s side, the queueing model can be described as *M/M/*1.

The following assumptions are made to describe our system.

1) Assumption 1. The instant user request arrival event constitutes a stationary Poisson process with the parameter λ.

2) Assumption 2. For each *AA*, the service time of different individual users are independent and identically distributed exponential random variables, in which the mean value is $1/\mu 1$.

3) Assumption 3. For *CA*, the service time of individual users are independent and identically distributed exponential random variables, in which the mean value is $1/\mu 2$.

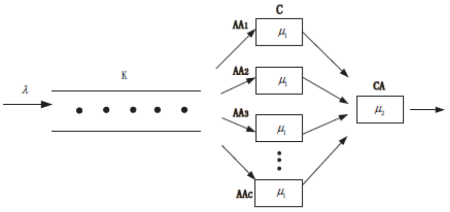

FIGURE 4.1:QUEUE MODEL WITH SINGLE-*CA*/MULTI-*AA*S

4.1.2 NUMERICAL EVALUATION

As we described above, when the queue model with multi-AAs is filled with N users, which means that K users are waiting in the queue and all C AAs are occupied, the newly arriving users are rejected. This means those new users would fail to get the secret keys. We analyze the probability of failure to show how to lower this failure rate with more AAs. Meanwhile, we analyze the average waiting time W′ q of users in our system. Based on the emulation of the scheme in [18], the average time of generating a secret key for an attribute is about 35ms (on 64-bit AMD 3.7 GHz workstation). Furthermore, we assume that users possess 10 attributes on average and the verification takes tenfold amount of time of that of the key generation. The parameters are set as: µ1 = 20/min, µ2 = 200/min, and K = 30. The performance analysis in terms of the average failure rate and the average waiting time is shown in the Figures respectively. In this shows the failure rate versus the arrival rate and the number of AAs. we can see that when the average failure rate of single-authority scheme is less than 5%, it can only support an arrival rate of less than 20/min. With increasing the number of AAs, the system can greatly increase its service capacity with the support of a greater arrival rate at the same failure rate. If we employ 7 AAs, the system can support the arrival rate of up to 150/min, with the failure rate of less than 5%. It is easy to infer that we can build our system based on the observation of key request rate, and then use an appropriate number of AAs to provide high equality service.

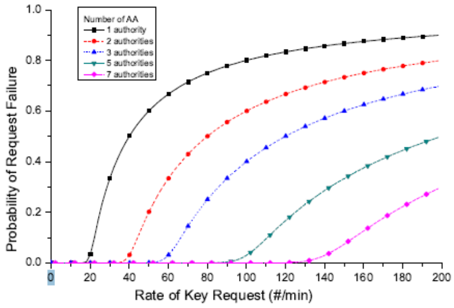

FIGURE 4.2 REQUEST FAILURE CHART

Ratio	No Of Centre Authority	No Of Attribute Authority Per 20 Clients	No Of Misbehaviours
Existing system	1	1	3
Proposed system	1	1	-
Previous system	1	0	-

FIGURE 4.4: PERFOMANCE COMPARISION TABLE

In this figure _1 = 20=*min*, _2 = 200=*min* and *K* = 30 shows the average waiting time versus the arrival rate and the number of *AA*s when $\mu 1$ = 20/*min*, $\mu 2$ = 200/*min*,*K* = 30.

Process	User authendication & file security(%)	Robustness(%)
Existing system(CP-ABE)	80	89
Proposed system(CBC)	99	96

FIGURE 4.3: PROCESS COMPARISION TABLE

From the figure, we can see that the average waiting time increases rapidly with the increase of arrival rate when the arrival rates are low. But later the average waiting time will become steady because newly arrival users will be rejected by the system due to the limit length of waiting queue. More specifically, with single *AA*, the average waiting time increases rapidly and reaches 1.5 *min*, which is unbearable. Whereas, with 7 *AA*s, the average waiting time is about 15*s*. Moreover, from Figure with the arrival rate less than 150, the failure rate is less than 5%. Although using more working *AA*s brings larger configuration cost, by combining the failure rate and the average waiting time, we can assure that the configuration of multiple *AA*s can provide secret key generation service with high quality as well as low cost.

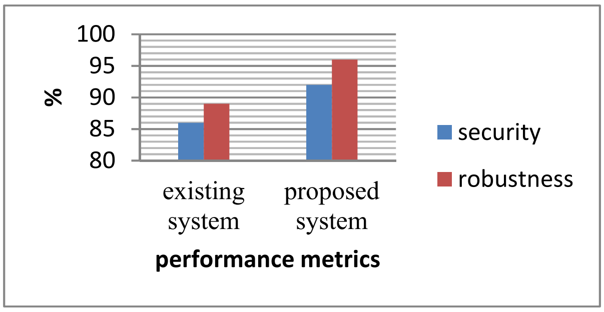

FIGURE 4.5: PERFORMANCE COMPARISION CHART

5. CONCLUSION

The proposed another system, to dispose of the single-point execution bottleneck of the current CP-ABE plans. By adequately reformulating Chain Block Cipher cryptographic system into our novel structure, our proposed conspire gives a fine-grained, powerful and proficient access control with one-CA/multi-AAs for open distributed storage.

Our plan utilizes numerous AAs to share the heap of the tedious authenticity confirmation and standby for serving fresh debuts of clients' solicitations. We additionally proposed a reviewing technique to follow a trait expert's potential rowdiness.

5.1 FUTURE SCOPE

Advance we consider the best execution in light of lining hypothesis demonstrated the prevalence of our plan over the customary CBC based access control plans for open distributed storage.

REFERENCES

[1] P. Mell and T. Grance, “The NIST definition of cloud computing,” *National Institute of Standards and Technology Gaithersburg*, 2011.

[2] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, “Enabling personalized search over encrypted outsourced data with efficiency improvement,” *IEEE Transactions on Parallel & Distributed Systems*, vol. 27, no. 9, pp. 2546–2559, 2016.

[3] Z. Fu, X. Sun, S. Ji, and G. Xie, “Towards efficient content-aware search over encrypted outsourced data in cloud,” in *in Proceedings of 2016 IEEE Conference on Computer Communications (INFOCOM 2016)*. IEEE, 2016, pp. 1–9.

[4] K. Xue and P. Hong, “A dynamic secure group sharing framework in public cloud computing,” *IEEE Transactions on Cloud Computing*, vol. 2, no. 4, pp. 459–470, 2014.

[5] Y. Wu, Z. Wei, and H. Deng, “Attribute-based access to scalable media in cloud-assisted content sharing,” *IEEE Transactions on Multimedia*, vol. 15, no. 4, pp. 778–788, 2013.

[6] J. Hur, “Improving security and efficiency in attribute based data sharing,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2271–2282, 2013.

[7] J. Hur and D. K. Noh, “Attribute-based access control with efficient revocation in data outsourcing systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 7, pp. 1214–1221, 2011.

[8] J. Chen and H. Ma, “Efficient decentralized attribute based access control for cloud storage with user revocation,” in *Proceedings of 2014 IEEE International Conference on Communications (ICC 2014)*. IEEE, 2014, pp. 3782–3787.

[9] M. Chase and S. S. Chow, “Improving privacy and security in multi-authority attribute-based encryption,” in *Proceedings of the 16th ACM conference on Computer and Communications Security (CCS 2009)*. ACM, 2009, pp. 121–130.

[10] W. Li, K. Xue, Y. Xue, and J. Hong, “TMACS: A robust and verifiable threshold multi-authority access control system in public cloud storage,” *IEEE Transactions on Parallel & Distributed Systems*, vol. 27, no. 5, pp. 1484–1496, 2016.

[11] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute based encryption for fine-grained access control of encrypted data,” in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS 2006)*. ACM, 2006, pp. 89–98.

[12] J. Bethencourt, A. Sahai, and B. Waters, “Ciphertextpolicy attribute-based encryption,” in *Proceedings of IEEE Symposium on Security and Privacy (S&P 2007)*. IEEE, 2007, pp. 321–334.

[13] J. Shao, R. Lu, and X. Lin, “Fine-grained data sharing in cloud computing for mobile devices,” in *Proceedings of 2015 IEEE Conference on Computer Communications (INFOCOM 2015)*. IEEE, 2015, pp. 2677–2685.

[14] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, “Securely outsourcing attribute-based encryption with checkability,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2201–2210, 2014.

[15] S. Ruj, A. Nayak, and I. Stojmenovic, “DACC: Distributed access control in clouds,” in *Proceedings of the 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2011)*. IEEE, 2011, pp. 91–98.

[16] K. Yang and X. Jia, “Expressive, efficient and revocable data access control for multi-authority cloud storage,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, 2013.

[17] J. Han, W. Susilo, Y. Mu, J. Zhou, and M. H. A. Au, “Improving privacy and security in decentralized cipher text policy attribute-based encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 665–678, 2015.

[18] T. Jung, X.-Y. Li, Z. Wan, and M. Wan, “Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 190–199, 2015.

[19] K. Yang, X. Jia, and K. Ren, “Secure and verifiable policy update outsourcing for big data access control in the cloud,” *IEEE Transactions on Parallel & Distributed Systems*, vol. 26, no. 12, pp. 3461–3470, 2015.