Analysis on Data Security with Time Constraint in Clouds

M. S. Patel[1] S. S. Londhe[2] S. D. Mulik[3] N. S. Shinde[4] V.V. Sawant[5]
[1,2,3,4]B.E Student [5]Guide
[1,2,3,4,5]Department of Computer Engineering
[1,2,3,4,5]SVPM's C.O.E. Malegaon (Bk.), 413115, Savitribai Phule, Pune University, Maharashtra, India

Abstract— Cloud computing is one of evolving technology nowadays, giving versatile services. But secure data sharing is susceptible in cloud computing environment. Full lifecycle privacy security is not implemented in Cloud, access control is challenging task to share sensitive data on cloud servers. One of novel approach for secure data self-destructing scheme is Key Policy Attribute Based Encryption with Time Specified attributes i.e. (KP-TSABE). The cipher text is labeled with time interval and private key is associated with particular time instant.KP-TSABE supports user defined authorization period by providing fine grained access control during the period. After User specified expiration time the data will be securely self-destructed.KP-TSABE scheme is secure under the decision l-bilinear Diffie-Hellman inversion assumption.

Key words: Sensitive Data, Secure Self-Destructing, Fine Grained Access Control, Privacy-Preserving, Cloud Computing

I. INTRODUCTION

With the speedy development of versatile cloud offerings, it becomes Associate in nursing increasing variety of liable to use cloud services to proportion facts during a crony circle within the cloud computing surroundings. as a result of it's not viable to place in effect complete life-cycle privacy security, get admission to manage becomes a tough endeavor, especially after we share sensitive info on cloud servers.

The shared information in cloud servers, however, sometimes contains user's sensitive info and desires to be protected. Because the possession of the info is separated from the administration of them, the cloud servers could migrate user's information to alternative cloud servers in outsourcing or share them in cloud looking out. Therefore, it becomes an enormous challenge to shield the privacy of this shared information in cloud, particularly in cross cloud and large information surroundings. So as to fulfill this challenge, it's necessary to style a comprehensive resolution to support user-defined authorization amount and to produce fine grained access management throughout this era. The shared information ought to be self-destroyed once the user outlined expiration time.

II. LITERATURE SURVEY

A. Attribute-Based Encryption

Attribute-based Attribute-based encoding is one among the important applications of fuzzy identification-primarily primarily based encoding [7]. ABE comes in favor's known as KP-ABE [8][11] and cipher text policy ABE (CP-ABE) [12][13]. In CP-ABE, the cipher text is related to the get entry to structure whereas the personal key carries a collection of attributes. Be then court docket et al. projected the primary CPABE theme [12], the disadvantage in their theme is that safety proof became handiest engineered underneath the well-known establishment version. To subsume this liability, Cheung et al. provided the other construction Beneath a classy model [13]. Waters used a linear secret sharing theme (LSSS) matrix as a most popular set of get entry to structures over the attributes associate degreed projected an economical and incontrovertibly comfortable CP-ABE theme to a lower place the standard version [14]. In KP-ABE, the construct is reversed: the cipher matter content consists of a collection of attributes and therefore the personal secret is expounded to the get entry to structure. The primary production of KP-ABE theme was projected in [8]. In their theme, once a user created a secret request, the relied on authority determined that mixture of attributes have to be compelled to appear inside the cipher matter content for the user to decode. instead of the employment of the Shamir mystery key technique [15] within the private key, this theme used an additional generalized form of secret sharing to place into impact a monotonic get right of entry to tree. Ostrovsky et al. provided the primary KP-ABE machine that supports the no monotone formulas in key rules [14]. Yu et al. used a combining technique of KP-ABE, proxy encoding, and lazy re-encryption which allows the records owner to delegate most of the computation obligations involved in fine-grained data access management to untrusted cloud servers while not revealing the underlying facts contents [13]. Tysowski et al. changed the ABE and leveraged re-encryption algorithmic rule to endorse a unique theme to protect mobile user's facts in cloud computing surroundings. Attributable to the shortage of your time constraints, the above-stated ABE schemes don't guide user-defined authorization period and comfortable self-destruction when expiration for privacy-maintaining of the records lifecycle in cloud computing

B. Secure self-destruction scheme

A noted technique for addressing this drawback is relaxed deletion of touchy statistics when expiration whereas the facts became used [15]. Currently, Cachin et al. employed a coverage graph to elucidate the link among attributes and therefore the protection magnificence and projected a coverage-based secure statistics deletion theme. Reardon et al. leveraged the graph construct, Btree form and key wrapping and projected a novel approach to the planning and analysis of comfortable deletion for persistent storage devices. Attributable to the homes of bodily garage media, the above-cited strategies are not applicable for the cloud computing surroundings because the deleted statistics could also be recovered only inside the cloud servers. A records self-destructing theme, 1st projected by means of Geambasuetal, could be a promising technique that styles a Vanish device permits customers to regulate over the lifecycle of the touchy facts. Wang et al. improved the Vanish device and projected a relaxed self-destructing theme for digital facts (SSDD). Within the SSDD theme, information is encrypted right into a cipher text that's then associated and

extracted to create it incomplete to face up to towards the standard cryptanalytics and therefore the brute-pressure attack. Then, each the decoding key and therefore the extracted cipher text area unit assigned into a distributed hash table (DHT) network to place Into impact self-destruction when the update length of the DHT network. However, Wolchok et al. created variety of experiments and confirmed that the Vanish machine is prone to Sybil attacks by the employment of the Vuze DHT community.

Therefore the security of the SSDD theme is likewise questionable. To deal with this problem, Zeng et al

C. *Time Specific Encryption*

A noted technique for addressing this drawback is relaxed deletion of touchy statistics when expiration whereas the facts became used [14]. Currently, Cachin et al. employed a coverage graph to elucidate the link among attributes and therefore the protection magnificence and projected a coverage-based secure statistics deletion theme. Reardon et al. leveraged the graph construct, Btree form and key wrapping and projected a novel approach to the planning and analysis of comfortable deletion for persistent storage devices [15]. Attributable to the homes of bodily garage media, the above-cited strategies are not applicable for the cloud computing surroundings because the deleted statistics could also be recovered only inside the cloud servers. A records self-destructing theme, 1st projected by means of Geambasuetal, is a promising technique that styles a Vanish device permits customers to regulate over the lifecycle of the touchy facts. Wang et al. improved the Vanish device and projected a relaxed self-destructing theme for digital facts (SSDD). Within the SSDD theme, information is encrypted right into a cipher text that's then associated and extracted to create it incomplete to face up to towards the standard cryptanalytics and therefore the brute-pressure attack. Then, each the decoding key and therefore the extracted cipher text area unit assigned into a distributed hash table (DHT) network to place into impact self-destruction when the update length of the DHT network. However, Wolchok et al. created variety of experiments and confirmed that the Vanish machine is prone to Sybil attacks by the employment of the Vuze DHT community. Therefore the security of the SSDD theme is likewise questionable. To deal with this problem, Zeng et al. projected a SeDas appliance that could be a singular integration of cryptographical techniques with active storage techniques. Xiong et al. leveraged the DHT network associate degreed identity-based altogether encoding (IBE) and projected an IBE-based comfortable self-destruction (ISS) theme. To be ready to guard the confidentiality and privacy protection of the composite files within the complete lifecycle in cloud computing, Xiong et al. applied the ABE algorithmic rule to suggest a comfy self-destruction theme for composite documents (SelfDoc). these days, Xiong et al. used identification-based altogether timed-launch encoding (identification-TRE) algorithmic rule [9] and therefore the DHT network and projected a full lifecycle privacy protection theme for sensitive facts (FullPP), that is capable of supply full lifecycle privateers safety for customers' touchy records with the help of creating it unclear previous a predefined time and robotically destructed when expiration [3]. The principle plan of the above-noted schemes is that they severally integrate specific cryptographical techniques with the DHT network to supply fine-grained data get admission to regulate throughout the lifecycle of the enclosed records and to place into impact records self-destruction when expiration. However, the usage of the DHT network can lead to the actual fact that the lifecycle.

III. CONTRIBUTION

In this paper, we advise a KP-TSABE scheme, that is a novel comfy self-destructing scheme for records sharing in cloud computing. We first introduce the perception of KP-TSABE, formalize the model of KP-TSABE and give the security version of it. Then, we give a specific creation technique about the scheme. Eventually, we prove that the KP-TSABE scheme is secure. Specially, KP-TSABE has the following advantages with regard to protection and fine-grained get admission to manage in comparison to other comfortable self-destructing schemes.

1) KP-TSABE supports the characteristic of user defined authorization length and ensures that the touchy information cannot be read each earlier than its preferred release time and after its expiration.
2) KP-TSABE does now not require the proper assumption of "No attacks on VDO earlier than it expires".
3) KP-TSABE is capable of put into effect fine-grained get admission to control during the authorization duration and to make the touchy information self-destruction after expiration without any human intervention.
4) KP-TSABE is validated to be secure beneath the usual version by way of the usage of the l-bilinear Diffie Hellman inversion assumption.

IV. SYSTEM MODEL

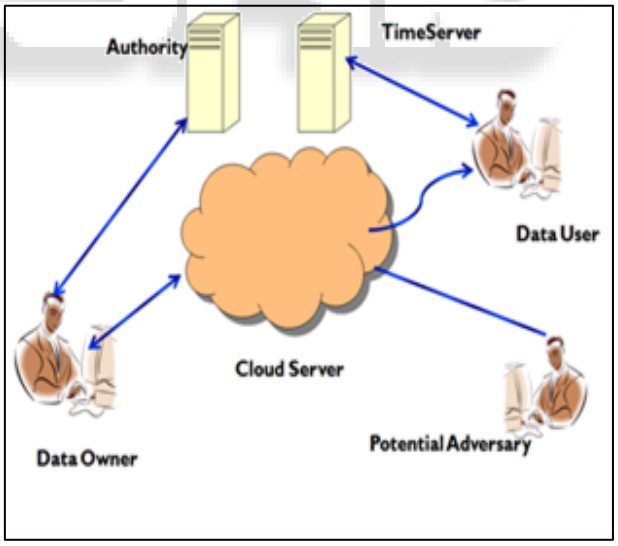

Fig. 1:

Data Owner: Data owner can provide data or files that contain some sensitive information, which are used for sharing with his/her friends (data-users). All these shared data are outsourced to the cloud servers to store.

Authority: It is an indispensable entity which is responsible for generating, distributing and managing all the private keys, and is trusted by all the other entities involved in the system.

Time Server: It is a time reference server without any interaction with other entities involved in the system. It is responsible for a precise release time specification.

Data Users: Data users are some peoples who passed the identity authentication and access to the data outsourced by the data owner. Notice that, the shared data can only be accessed by the authorized users during its authorization period. Cloud Servers. It contains almost unlimited storage space which is able to store and manage all the data or files in the system. Other entities with limited storage space can store their data to the cloud servers.

V. ALGORITHM

Taher ElGamal given a cryptosystem that's predicated on the distinct log draw back mentioned at intervals the last section [6]. It depends on the thought that the metric capacity unit cannot be found in potential time, whereas the inverse operation of the power area unit typically computed expeditiously. The initial public key system planned by Diffie and playwright wants interaction of each parties to calculate a regular personal key. This poses problems if the cryptosystem ought to be applied to communication systems where every party do not appear to be able to act in reasonable time as a result of delays in transmission or inconvenience of the receiving party.

Thus ElGamal simplified the Diffie-Hellman key exchange formula by introducing a random exponent k. This exponent is Associate in nursing replacement for the private exponent of the receiving entity. Attributable to this simplification the formula area unit typically accustomed inscribe in one direction, whereas not the necessity of the half to need actively half. The key advance here is that the formula area unit typically used for secret writing of electronic messages, that area unit transmitted by the suggests that of public store-and-forward services. During this section, the ElGamal cryptosystem area unit introduced to the reader.

A. Key Generation

Participant generates the public/private key pair

$$KeyGen(OK_{pri}, UK_{pri})$$

OKpri is Owner private key.
UKpri is User private key.
OKpub is Owner public key.
UKpub is User public key.
Ek is encryption key.
g is primitive root or generator.

B. Encryption Procedure

$$Encrypt(E_k, M)$$

Where, M is message to be encrypted.
CT is cipher text.

$$CT = E_k \bmod h$$
$$O = \{CT\}$$

C. Decryption Procedure

$$Decrypt(CT, UK_{pri})$$
$$E_k OK_{pub}^{UK_{pri}} \bmod h$$
$$E_k . x \bmod h = 1$$

X is number such that after computation answer is 1.

$$M = x . CT \bmod h$$
$$O = \{M\}$$

VI. ANALYSIS

The KP-TSABE theme is verified to be secure under the quality model. Therefore, we tend to consistently compare this theme with the prevailing self-destruction solutions (e.g., Vanish, SSDD, ISS, and FullPP [3]) from the subsequent aspects, e.g., requirement condition, algorithm, resistance on attacks, fine-grained access management, user-defined authorization amount, etc. The results of the great comparison are shown in Table one.

Security Propertise	Vanish	SSDD	ISS	Full PP	KP_TBASE
Need "no attacks on VDO before it expires"?	YES	YES	YES	NO	No need
Leveraging what kind of algorithm ?	Symmetric	Symmetric	IBE	ID-TRE	KP-TSABE
Whether ciphertext is destructed or not?	NO	YES	YES	YES	No need
Whether the key is destructed or not?	YES	YES	YES	YES	No need
Resistance on the traditional cryptanalysis?	NO	YES	YES	YES	YES
Resistance on the Sybil attacks?	NO	NO	YES	YES	-
Resistance on the collusion attack?	-	-	-	-	YES
Supporting fine-grained access control?	NO	NO	YES	YES	YES
Providing full lifecycle privacy protection ?	NO	NO	NO	YES	YES
Supporting user-defined time intervals?	NO	NO	NO	Half	YES

Security proof under standard model?	NO	NO	NO	YES	YES

Table 1:

All the schemes of Vanish, SSDD and ISS want the best assumption "no attacks on VDO before it expires". Since a Sybil antagonist is in a position to crawl sufficient key shares from the DHT network to reconstruct the decryption key. Once the antagonist gets the VDO from the cloud servers before it expires, he/she can decrypt it with the reconstructed decipherion key to obtain the plaintext. FullPP [3] doesn't want this ideal assumption as a result of the decoding key is encrypted by the ID-TRE rule. Though the adversary crawls sufficient key shares from the DHT network, he cannot reconstruct the decryption key since he does not have the ID-TRE private key. KPTSABE also does not need the ideal assumption because it does not require the DHT network. Algorithm and resistance on attacks. Since both Vanish and SSDD only use symmetric encryption to encrypt the sensitive message, they bring complex key management and cannot achieve fine-grained access control for different users with different attributes. Vanish sends the entire ciphertext to the cloud server, so it cannot resist against the traditional cryptanalysis. Since the SSDD scheme distributes a part of the ciphertext and the decryption key to the DHT network, both of which will be self-destructed after expiration, so the cloud server stores incomplete cipher text. Therefore, SSDD can resist against the traditional cryptanalysis. However, Vanish and SSDD cannot resist against the Sybil attackers who can continually crawl the key shares from the DHT network to recover the decryption key. In contrast, each ISS and FullPP [3] will not solely resist against the normal cryptography and the Sybil attacks however additionally implement versatile access management owing to the IBE and ID-TRE algorithms. KP-TSABE doesn't have the matter of the Sybil attacks as a result of there's no use of the DHT network. What is more, it will offer fine-grained access management through combining completely different attributes with variance time intervals. User-defined authorization amount. Vanish, SSDD, ISS and FullPP [3] leverage the DHT network to store the key shares or the hybrid ciphertext shares, that area unit self-discarded by the DHT nodes once an amount of your time. That the expiration time is proscribed by the update amount of the DHT network and it can't be controlled by the sensitive data owner. Higher than those schemes, within the KPTSABE scheme, each attribute within the attribute set associated with the cipher text is matched with a time interval, that is that the authorization amount of the sensitive information and is predefined by the information owner.

Therefore, the authorization amount and therefore the expiration time don't seem to be restricted by the system constraint, but flexibly to be outlined by the owner. Security proof. Vanish, SSDD, and ISS don't offer the protection proof. The ID-TRE in the FullPP theme is verified to be secure below the additive Diffie-Hellman (BDH) assumption. Furthermore, the KP-TSABE theme is verified to be secure below the quality model with the choice l-Expanded B DHI assumption to resist against the traditional cryptography and therefore the collusion attack. In conclusion, the KP-TSABE theme is superior to the prevailing self-destruction solutions from several security properties.

The Execution times for every the Elgamal and RSA algorithms area unit shown on the Tables and Figures. The day's area unit measured in milliseconds, but regenerate to seconds as displayed on the result templates. we have a tendency to tend to watch and deduce as follows from the results obtained.

In the secret writing and communication methodology, the RSA performs higher than Elgamal altogether cases. In the secret writing methodology, the Elgamal outperforms RSA; meaning that text messages area unit decrypted faster by Elgamal than can the RSA technique. At intervals the signature verification methodology, the RSA once more performs over the Elgamal approach. once viewed together tool, the RSA is superior to the Elgamal formula in terms of method speeds. This, in part, explains why the RSA formula has been and remains getting used in turning out with many security protocols for information communication.

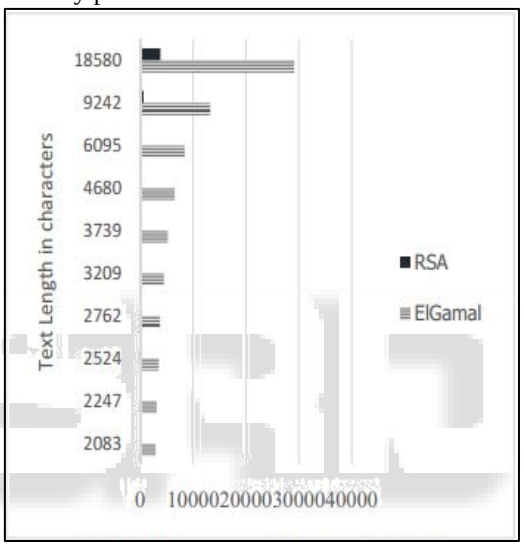

Chart. 1: Execution Time for Encryption and Signing

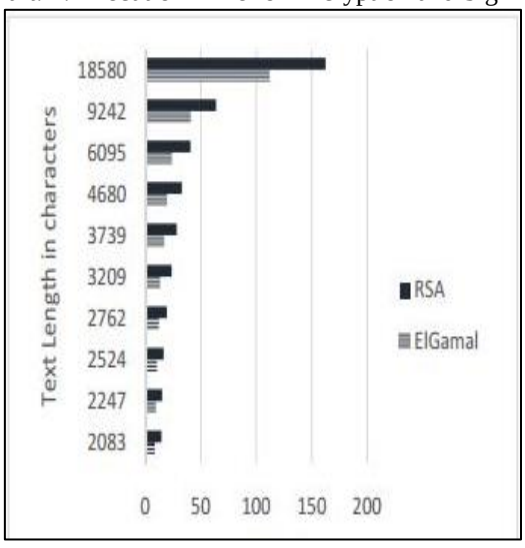

Chart. 2: Execution time for Decryption

REFERENCES

[1] B. Wang, B. Li, and H. Li, "Oruta: Privacy-preserving public auditingfor shared data in the cloud," Cloud Computing, IEEE Transactions on, vol. 2, no. 1, pp. 43–56, 2014.

[2] J. Xiong, Z. Yao, J. Ma, X. Liu, Q. Li, and J. Ma, "Priam: Privacy preserving identity and access management scheme in cloud," KSII Transactions on Internet and Information Systems (TIIS), vol. 8, no. 1, pp. 282–304, 2014.
[3] J. Xiong, F. Li, J. Ma, X. Liu, Z. Yao, and P. S. Chen, "A full lifecycle privacy protection scheme for sensitive data in cloud computing," Peerto-Peer Networking and Applications. [Online]. Available: http://dx.doi.org/10.1007/s12083-014-0295-x
[4] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: A systematic review," Cloud Computing, IEEE Transactions on, vol. 1, no. 2, pp. 142–157, 2013.
[5] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," Network, IEEE, vol. 28, no. 4, pp. 46–50, 2014.
[6] X. Liu, J. Ma, J. Xiong, and G. Liu, "Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data," International Journal of Network Security, vol. 16, no. 4, pp. 351–357, 2014.
[7] Sahai and B. Waters, "Fuzzy identity-based encryption," in Advances in Cryptology–EUROCRYPT 2005, ser. LNCS, vol. 7371. Springer, 2005, pp. 457–473.
[8] 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. ACM, 2006, pp. 89–98.
[9] F. Chan and I. F. Blake, "Scalable, server-passive, useranonymous timed release cryptography," in Proceedings of the International Conference on Distributed Computing Systems. IEEE, 2005, pp. 504–513.
[10] K. G. Paterson and E. A. Quaglia, "Time-specific encryption," in Security and Cryptography for Networks. Springer, 2010, pp. 1–16.
[11] Q. Li, J. Ma, R. Li, J. Xiong, and X. Liu, "Large universe decentralized key-policy attribute-based encryption," Security and Communication Networks, 2014. [Online]. Available: http://dx.doi.org/10.1002/sec.997
[12] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proceedings of the 28th IEEE Symposium on Security and Privacy. IEEE, 2007, pp. 321– 334.
[13] L. Cheung and C. C. Newport, "Provably secure ciphertext policy abe," in Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007, pp. 456– 465.
[14] Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," Public Key Cryptography–PKC 2011, pp. 53–70, 2011.
[15] Shamir, "How to share a secret," Communications of the ACM, vol. 22, no. 11, pp. 612–613, 1979.

