International Journal of Engineering Research ISSN:2319-6890)(online),2347-5013(print)
Volume No.5 Issue: Special 4, pp:790-991 20 May 2016

SERBAC Framework to Prevent Data Integrity in Cloud Structure

Sunitha B S
Associate Professor, ISE Department, EPCET
PhD Research Scholar, VTU
Bangalore, INDIA
sunitha74123@gmail.com

Dr. Anirban Basu
Professor, CSE Department,
APSCE
Bangalore, INDIA
anirban@pqrsoftware.com

***Abstract*— Cloud computing and its services are rapidly developing; there is a growing trend for bulk storage of data in the cloud. This result leads to an important security issue of controlling and preventing the confidential data stored in the cloud. However, many of existing approaches have a major shortcoming, as they assume the server is trustworthy and require complete disclosure of sensitive location information from the cloud user. In this work we describe a new method for completely distributed authentication using a cloud based framework for Session Authentication ticket. In our proposed scheme, the authentication takes place for every user with a role along with a request. The session authenticator service provider verifies the tokens and uses dual encryption to evaluate the roles of the user thus providing the efficiency. In this paper, we define the protocols based on the data preserving integrity of data and the dual encryption mechanism based on our efficient access control system SERBAC.**

Keywords—Cloud ACR, group key management, Session Authentication Ticket

I. INTRODUCTION

Cloud computing is one of the profitable developments for business, cost effective, virtualization of services that once required expensive and local hardware. Cloud computing, likewise gives different services to cloud users to build, deploy and manage their applications on the cloud. Cloud includes virtualization of assets that keeps up and oversees without anyone else's input. The fundamental question is the manner by which do we shield information furthermore security from being traded off. Security is principally fundamental for solid protection in all web based components, yet security alone is insufficient. Security and expense are the top issues in this field and they change incredibly, contingent upon the vendor one choice. In spite of the success and recognition of the cloud computing model and the broad accessibility of service providers and tools, various challenges and risk factors innate to this new model of security services in cloud computing. A way to deal with mitigating these concerns is the utilization of encryption. Despite the fact that, encryption guarantees the confidentiality of the information against the cloud, the utilization of conventional encryption methodologies is not adequate to bolster the requirement of fine-grained organizational access control roles (ACRs). Our framework depends on enhancing collaboration effort between cloud providers and cloud users in dealing with the security of the cloud platform and the hosted services. Accordingly, to it the session authentication is proposed. In this proposed Session Authentication service, a double encryption is taken into consideration after which is a validated authenticated system. To exhibit this framework, two encryption strategies have been proposed. So for every cloud user in the SERBAC, the encryption happens from the session validation server, which is more effective contrasted with the other encryption systems.

II. RELATED WORK

Much work has been done in the preserving data Integrity in Cloud Computing. Let us investigate some of the surveys which exists. In the existing work, there are many hierarchy access control schemes [22] [3] which have been developed based on hierarchical key management schemes, and new approaches using hierarchical key management schemes to enforce roles based on access control for data storage are discussed in [7] [10].However, these solutions, also [8] have various limitations. For instance, if there is a many number of data owners and users involved, setting up the key infrastructure is the overhead can be very high indeed. Furthermore, on revoking user's access permission all the keys known to users as well as all the public values related to these keys needs to be updated, which make these schemes impractical. One more approach for the management of keys is Hierarchical ID-based Encryption, such as [12], [13]. However, in a Hierarchical ID-based Encryption scheme, with growth in the depth of hierarchy, identity length becomes longer.

In one more survey, encryption scheme based on attribute is proposed [11] based on the work in [14], and some other attributes-based encryption schemes have been proposed afterwards. In these schemes, data is encrypted to a set of attributes, and users who have the private keys associated with these attributes can decrypt the data. Existing works have provided an alternative approach to preserve integrity of the data stored in a distributed environment using a various access control mechanism, such as [9]. In [15], it is shown that an attribute-based encryption scheme can be used to enforce access control of roles. However, in that approach, the user key size is not constant, and the revocation of a user will result in a key update of all the other users of the same role. [16] Also investigated the solutions of using attribute-based encryption scheme in access control model. However, existing solution only maps the attributes to the role level in controlling the access [1], and they assumed that the access control system itself would determine the membership of the user. Even though these attribute-based constructs are provably secure, but they are not designed for group management and especially in supporting forward security when a user revokes from the group and in providing backward security when a new user joins the group. Other approaches to protect data privacy in a cloud environment include using direct encryption and proxy re-encryption. In these cryptographic schemes, data is allowed to be encrypted

directly to the users with whom the owners wish to share the data [17], [18]. This is analogous to the access control roles in Discretionary Access Control (DAC) model. Hence they are usually used in systems where the DAC model is adopted. Since the permissions in such systems are specified either in a flat out structure or in an access matrix, we do not compare them with our schemes as the access roles are specified differently in SERBAC model. Recent research efforts, [21] [4] have proposed approaches to construct a privacy preserving access control systems using a third-party storage service. In such approaches, the data owner has to enforce the access control of roles and the privacy of the users from the content publisher is not protected.

Further, in some approaches [5] [6], multiple encryptions of the same document are required which is inefficient. Recently Liang et al. [25] has extended the traditional PRE to attribute based systems and independently Chu et al. [25] has extended the traditional PRE to support conditions where a proxy can re-encrypt only if the condition specified by a public key to a third party is satisfied. However, they do not protect the identity attributes of the users who access the system [2] and are difficult to manage. LAN Zhou et al [23] present a design of a trust-based cloud storage system, which shows how the trust models can be integrated into a system that uses cryptographic RBAC schemas. Wenhui Wang et al [24] paper suggests an adaptive access algorithm by introducing the trust into cloud computing to decide the access control to the resources using an improved RBAC technique to solve more complex and difficult problems in the cloud computing environment. Jiangfeng Li et.al [25] proposed 4D-role based multitenancy model, for running various services and applications on the multitenancy cloud platform. To overcome these issues in the existing work, we have proposed a novel approach to preserve data Integrity which is secure and accessible in the public cloud. A highly secure performance role access control evaluation mechanism is proposed based on *SERBA*.

III. Proposed Work

Security Initialization plans and the secure tunnelling mechanism are discussed here. Abstract view of the main algorithms of those protocols and how it is used to build the privacy-preserving attributes based group key management in the data integrity preservation scheme are briefed. And afterward, a review of securing the roles and the double encryption security instrument approach in SERBAC taking into account the Session administration is given.
and Acronyms

A. Security Initializations

It takes care of the issue of how efficiently to encrypt a message and sends it across to a subset of the cloud users with a framework. The subset of cloud users can change dynamically. The cloud users are called privileged users, in the broadcast encryption and the unauthorized users called revoked. We indicate the set of cloud users by C, the collection of revoked clients V. The set of cloud users who had privilege to access the cloud, is subsequently, C⁄V. We set A= |C| and v=|V|. The broadcast encryption takes after the procedure where the message for each privileged cloud user is encrypted independently and afterward broadcast all the encrypted messages which is exceptionally inefficient since the message length is substantial which is given by (O (A-v)). Along these lines subsets-cover algorithm that supports broadcast encryption with stateless users is utilized. The calculation assembles a binary tree and assigns cloud users to the leaf nodes and in this way brings about a predefined user group. Each such group is known as a subset. A user can be an individual from a few subsets. The cover, meant by R, is defined as the set of subsets that contains all the privileged users, that is, cloud users in C ⁄ V. The subsets in the cover are disjoint and consequently each special user has belongs to only one subset.

A subset-cover based broadcast encryption is built up taking into account the following algorithms: Setup, GetSecKeys, GetCover, Broadcast, KeyDer and Decrypt. Each of the algorithms is characterized as follows. Setup (b,L): The server builds a binary tree T where there are at least L leaf nodes for the security parameter b which means the bit length. Every node in T is either allocated a unique key whose length is chosen by b, or can computationally infer a unique key. The cloud user, $_i$=1, 2,…, L, is assigned Leaf node. GetSecKeys (c_i): The server gives all the key allowed to c_i in T. GetCover (C ⁄ V): Given the privileged user set C ⁄ V, the server yields the cover R, that is, the arrangement of disjoint subsets that cover all the privileged cloud users. Broadcast (M,R): The server creates a session key Y and encrypt the message G with Y and encrypt Y with every key in the cover R. KeyDer (, R): The cloud user c identifies its subset in the cover R, yields the key that decrypts the session key. Decrypt ( ): It decrypt the encrypt message M with the key Y, to yield the message G. Here, we consider the complete sub tree algorithm. The complete sub tree algorithm enhances the fundamental strategy for at the same time revoking v cloud users and depicting the privileged users utilizing v ,log(A ⁄ v) subsets.

B. Secure tunneling mechanism

It negligently conveys a message to the user who fulfill certain conditions. This protocol comprises of three substances: a server S, a cloud client C and a trusted third party called the provider P. The tunneling Mechanism is set up in view of the algorithm: Setup, GenCom and GetData. This algorithm is as given below.

Setup (b): The P runs a Pedersen commitment setup convention to produce the framework parameters, a finite cyclic gathering G of huge prime request m, two generators x and y of G. The measure of m is reliant on the security parameter b. GenCom(a): A C needs to commit on the worth a. It presents a to the P. The P registers the Pedersen commitment $t = x^{'}$. Randomly chosen from . The P

digitally signs t and sends v, t and the signature of t to the C. *GetData* (t, con): The C sends the signed commitment t and shows the S's condition cc that it needs to fulfil. cond has the format "name predicate esteem" where the predicate can be $\geq, >, \leq, <$ *or*. After an interactive session, the S encrypts the information r and sends the encrypted information, called envelope, to the C. The C can decrypt and access data on satisfying the condition. The properties below mentioned are carried by the OCBE conventions. The does not learn the identity attributes of the users. A can open the envelope only if it's committed attribute value satisfies the condition. A cannot submit fake commitments in order to satisfy a condition as the commitments are signed by the .

C. Secure Data Integrity Preservation

Here we utilize the Broadcast Group Key Management and Privacy Preserving Attribute Based-Group Key Management protocols. The general development depends on the Attribute Based-Group Key Management plan which is an expressive build of the access control vector Broadcast Group Key Management scheme. The Broadcast Group Key Management expands the Group Key Management where the rekey operation is performed with a single broadcast show without requiring the utilization of private communication channels. The Broadcast Group Key Management plans don't give users the private keys, rather users are given a secret key which is consolidate with public data to get the actual private keys. Such schemes have the benefit of requiring a private communication once for the underlying secret sharing. The consequent rekeying operations are performed utilizing one broadcast message. Further, in such schemes accomplishing forward and in reverse security requires just to change the public data and does not influence the secret shares given to existing users. The foundation of Broadcast Group Key Management plan comprises of five algorithms: Setup, SecGen, KeyGen, KeyDer, and ReKey. The foundation of Attribute Condition is as per the following. An attribute condition A is a statement of the structure: 〖iden〗 attr qt where name is the name of a identity property attr, q is a relational operator, for example, $=,>,\geqslant,<,\leqslant,\neq$ and t is a quality that can be expected by the characteristic attr. The foundation of Access Control Roles is as per the following. An entrance control parts ACR is a set (e,d). Where, d signifies set of data items $\{D_1,. ,D_t\}$ and e is a monotonic expression over an set of attribute conditions that must be satisfied by a C to have entry to d.

The ACR is encapsulated in an access structure A. A will be a tree in which the internal nods represent threshold gates and the leaves depicts broadcast Group Key Management instances for the attributes. The objective of the access tree is to permit the inference of the gathering key for just the users whose attributes fulfill the access structure A. Every threshold gate in the tree is describes by its child nodes and an threshold value. Because of space constraints , the abstract algorithm of the Privacy Preserving Attribute Based-Group Key Management is given. In this manner, the Privacy Preserving Attribute Based-Group Key Management is set up with the algorithm: Setup, SecGen, KeyGen, KeyDer and ReKey. Setup (p,S,S_a): It takes the security parameter p, the maximum group size S, and the attribute conditions S as information, introduces the framework. SecGen(β): The secret generation algorithm gives a , $1 \leqslant i \leqslant S$ an set of secrets for each commitment 〖com〗_a∈ β , , $1 \leqslant i \leqslant m$. It summons Security Initialization::GetSecGen and Secure Tunneling Mechanism::GetData calculations. KeyGen(ACR): The key generation algorithm access the access control parts ACR as the input and yields a symmetric key Y, an set of public data set μ and an access tree A . It invokes Security Initialization::GetCover() and Secure Tunneling Mechanism::KeyGen algorithm. KeyDer(α,μ,A): Given the set of identity attributes α, the set of public information set μ and the access tree A, the key derivation algorithm outputs the symmetric Y only if the identity attributes in α satisfy the access structure A . It invokes Security Initialization::KeyDer and Secure Tunneling Mechanism::KeyDer algorithms. ReKey (ACR): The rekey algorithm is similar to the KeyGen algorithm. It is executed whenever the dynamics in the system change.

IV. PERFORMANCE EVALUATION

The Data Integrity Preservation "SERBAC" model has been developed for highly competitive and secure cloud computing environment. The Visual Studio 2010 framework 4.0 with C# had been used to present system model . The overall system has been developed and implemented with Amazon S3 public cloud platform. Simulation is carried over the developed system with various performance parameters like overhead in role computation, user creation and storage overhead for the user based on the Session based Authentication. The relative study for various factors has been performed against the existing mechanisms. This system or model performance has been verified for various user sizes with dynamic role assignments and the relative throughput as well as performance parameters have been checked for its robustness justification. Here, the decomposition of role takes place and integrity of data is preserved. The dual encryption mechanism is been processed for every user. We have created the system where the RuleSet can have the maximum number of 8. Based on this RuleSet, various roles and users can be created for the performing the task. Each users are assigned certain set of rules. Then, the Session authentication service takes place for each and every user, preserving the data.

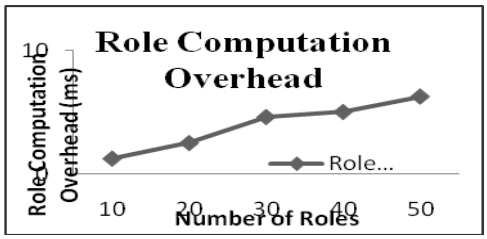

Figure 1: Role Computation Overhead

ISSN:2319-6890)(online),2347-5013(print)
20 May 2016

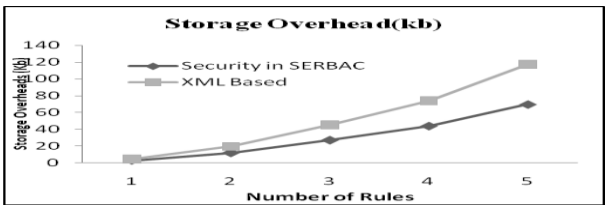

Figure 2: Storage Overhead

This above graph (Figure 2) is plotted making the comparison of the storage overhead of our proposed system security in against the existing system [9]. From the figure it is clear that the storage overhead is better even though the number of roles is increased. In the existing system, the xml sheets are created for each and every user and also for the permission of roles where it consumes lot of storage as well as the computation time. Whereas in our proposed system, the RuleSet for the multiple roles are already been computed based on the binary format of request and then it is assigned to the users which results in extreme minimization of storage space. Hence, the secureness is predicted based on the encryption which takes place twice. For each and every user, the Session Authentication takes place in which it verifies the tokens and evaluates the roles.

V.CONCLUSION

The insights of security and how to access to Cloud Computing are discussed in this paper. The popularity of cloud based services and their wide usage by enterprises and government organization. Cloud service providers still lack security services that guarantee consistency of both data and access control, among the various data centers. Our proposed paper, session authentication SERBAC based dual encryption mechanism. Our experimental results show that this method is secure, efficient and feasible which consumes less time and storage capacity even though permitting large number of users and roles and importantly more secure with the Session Authentication. Finally, we conduct comprehensive performance analysis, which shows that our framework is more secure, efficient and practical than existing schemes.

REFERENCES

[1] Min Xu, Duminda Wijesekera, Senior Member, IEEE, and Xinwen Zhang, Member, IEEE Runtime Administration of an RBAC Profile for XACML. Dec. 2011

[2]Mohamed Nabeel, Member, IEEE, Ning Shang, and Elisa Bertino, Fellow, IEEE Privacy Preserving Policy-Based Content Sharing in Public Clouds, Nov. 2013

[3]H. R. Hassen, A. Bouabdallah, H. Bettahar, and Y. Challal, "Key management for content access control in a hierarchy," Computer Network , vol. 51, no. 11, pp. 3197–3219, 2007

[4]S. Coull, M. Green, and S. Hohenberger, "Controlling Access to an Oblivious Database Using Stateful Anonymous Credentials," Proc. 12th Int'l Conf. Practice and Theory in Public Key Cryptography, pp. 501-520, 2009.

[5]J. Camenisch, M. Dubovitskaya, and G. Neven, "Oblivious Transfer with Access Control," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp. 131-140, 2009.

[6]K.P.N. Puttaswamy, C. Kruegel, and B.Y. Zhao, "Silverline: Toward Data Confidentiality in Storage-Intensive Cloud Applications," Proc. Second ACM Symp. Cloud Computing (SOCC '11), pp. 10:1-10:13, 2011.

[7]S. D. C. Di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in Proc. VLDB, Sep. 2007, pp. 123–134.

[8]P. Samarati and S. D. C. Vimercati, "Data protection in outsourcing scenarios: Issues and directions," in Proc. ASIACCS, Apr. 2010, pp. 1–14.

[9]S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc. IEEE INFOCOM, Mar. 2010, pp. 534–542.

[10]C. Blundo, S. Cimato, S. D. C. Di Vimercati, A. D. Santis, S. Foresti,S. Paraboschi, et al., "Efficient key management for enforcing access control in outsourced scenarios," in SEC (IFIP), vol. 297. New York, NY, USA: Springer-Verlag, May 2009, pp. 364–375.

[11]V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. ACM Conf. Computer. Communication . Sec., Oct./Nov. 2006, pp. 89–98.

[12]C. Gentry and A. Silverberg, "Hierarchical ID-based cryptography," in ASIACRYPT (Lecture Notes in Computer Science), vol. 2501. New York, NY, USA: Springer-Verlag, 2002, pp. 548–566.

[13]D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical identity based encryption with constant size ciphertext," in EUROCRYPT (Lecture Notes in Computer Science), vol. 3494. New York, NY, USA: Springer Verlag, May 2005, pp. 440–456.

[14]A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Proc. EUROCRYPT, 2005, pp. 457–473

[15]L. Zhou, V. Varadharajan, and M. Hitchens, "Enforcing role-based access control for secure data storage in the cloud," Computing . J., vol. 54, no. 13, pp. 1675–1687, Oct. 2011.

[16] Y. Zhu, D. Ma, C. Hu, and D. Huang, "How to use attribute-based encryption to implement role-based access control in the cloud," in Proc. Int. Workshop Sec. Cloud Computing ., 2013, pp. 33–40.

[17] E.-J. Goh, H. Shacham, N. Modadugu, and D. Boneh, "SiRiUS: Securing remote untrusted storage," in Proc. NDSS, 2003, pp. 1–15.

[18]G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in Proc. NDSS, Feb. 2005, pp. 29–43.